Submitted to *INFORMS Journal on Computing*
manuscript (Please, provide the mansucript number!)

Learning-empowered Benders Decomposition for Flow Hub Location in E-Commerce

Tao Wu
School of Economics & Management, Tongji University, Shanghai 200092, China, danielwu9999@gmail.com

Weiwei Chen
Department of Supply Chain Management, Rutgers University, Piscataway, NJ 08854, USA, wchen@business.rutgers.edu

Jean-François Cordeau, Raf Jans
Department of Logistics and Operations Management, HEC Montréal, H3T 2A7, Canada, jean-francois.cordeau@hec.ca, raf.jans@hec.ca

This paper studies a flow hub location problem (FHLP) stemming from recent trends in network design for e-commerce businesses. Specifically, e-commerce companies are flexible and agile in re-optimizing their logistics networks, including supplier (origin) and customer zone (destination) decisions. Furthermore, a large number of commodities (flows) and a relatively small sales volume for each product incentivize e-commerce retailers to lease warehouse spaces as hubs, yielding a large number of hub location candidates. As such, the proposed FHLP determines the origin and destination of each flow simultaneously with the hub location and flow routing decisions, in contrast to the classical hub location problems where the origins and destinations of all flows are predetermined. To solve this large-scale optimization problem, we propose an optimization algorithm that combines Lagrangian relaxation and Benders decomposition. Novel acceleration techniques, such as a clustering-empowered multi-commodity Benders reformulation, learning-empowered elimination tests, and variable reduction techniques, are further developed to improve the performance and convergence of the algorithm. The efficiency of the proposed algorithm is evaluated via extensive computational experiments. The computational results show that, when compared to three other benchmark methods, the proposed algorithm can achieve optimal solutions much quicker for small-sized test instances and reduce optimality gaps from 13-26% to 5-6% for large-sized test instances with 600-700 nodes.

Key words: E-commerce, hub location, machine learning, mixed-integer linear programming, Benders decomposition

1. Introduction

Hub location problems (HLPs) are an important subset of classical facility location problems (Farahani et al. 2013) and lie at the heart of network design decisions. They encompass a challenging class of optimization problems that determine the choices of hubs and the interconnections between them to optimize selected efficiency metrics. More specifically, HLPs have natural links to various applications and prevail in planning activities for supply chain networks, logistics and transportation networks, computer networks, and telecommunication systems. Hubs in these networks serve as centralized handling and sorting, consolidating, connecting, and dispatching points for flows going through them. In addition to selecting a subset of hubs to operate from a given set of candidates, HLPs also decide the optimal links that connect

the selected hubs as well as the routing of flows in the network from the origins to the destinations. As a result, hub location decisions typically aim to minimize the total costs associated with setting up the hubs, as well as the hub operating costs, such as transshipment costs between hubs, origins, and destinations. Alternative metrics like delivery speeds and environmental impacts may also be considered in single or multiple-criteria decision models.

Despite being a classical problem in the operations research literature, HLPs have attracted great attention in recent supply chain practices. Several factors have forced companies to reconsider and restructure their supply chain networks. One prominent factor is the COVID-19 pandemic that has significantly altered customer behaviors, forcing retailers with brick-and-mortar stores to rethink their strategic locations. The turmoil caused by the pandemic has also exposed vulnerabilities in supply chains. As such, companies started paying more attention to resilience in their supply chains and will continue to enhance the network structure, hub locations, and connectivity between hubs. Further, it has been reported that there is a warehouse shortage across the United States with nearly 96% of existing commercial warehouse space already in use[1], driving up the rental prices for warehousing and storage. All these factors have triggered the need to re-optimize logistics networks and hub locations to remain competitive and cost-efficient.

In this paper, we focus on an extended HLP stemming from recent trends in network design for e-commerce businesses. Indeed, e-commerce has been booming due to its convenience to end customers. Customers can now view, compare, and order products at their fingertips on computers and mobile devices, avoiding the hassle of traveling to local stores. Meanwhile, sellers have also benefited from the flexibility provided by e-commerce. It reduces the barrier of working capital requirements to build and maintain expensive brick-and-mortar stores. Sellers can simply set up a website, and ship the products directly to the consumers from their warehouses (i.e., hubs) once an order is received. The COVID-19 pandemic has further accelerated the adoption of e-commerce around the world. For example, in the U.S., the e-commerce sales saw a 14.2% increase in 2021 after a rapid 50.5% increase in 2020[2]; the e-commerce revenue in Europe grew by 13% in 2021, where 73% of European internet users shopped online[3]; and China has become the largest e-commerce market globally[4].

The growth of these direct-to-consumer channels has transformed the order fulfillment operations and requirements, calling for an extended HLP to be solved. First, small to medium-sized online retailers often sell a large number of products with a relatively small sales volume for each product. Their customer demands and customer bases also change more often than traditional retailers. As a result, they are in favor of leasing hubs or warehouse spaces (e.g., from third-party logistics) (Gelareh et al. 2015, Wu et al. 2021) rather than building and owning an entire warehouse. They are also much more flexible and agile in

[1] See https://www.cnbc.com/2021/11/29/online-shopping-black-friday-cyber-monday-warehouses.html.

[2] See https://www.forbes.com/sites/jasongoldberg/2022/02/18/e-commerce-sales-grew-50-to-870-billion-during-the-pandemic.

[3] See https://ecommercenews.eu/european-ecommerce-grew-13-in-2021.

[4] See https://www.trade.gov/country-commercial-guides/china-ecommerce.

shifting their hub locations in response to changing products and demands. Therefore, the hub candidates in the problem under study can be very large, rendering the optimization problem even more challenging to solve than traditional ones. Second, traditional HLPs come with predetermined origins and destinations. In contrast, the HLPs studied in this paper determine the origin and destination of a flow (i.e., product) from a set of alternatives, in addition to the intermediate hub locations and flow routing. Particularly, the origin and destination candidates are often subject to capacity limitations. For the choice of origin, online retailers have the flexibility of outsourcing the procurement of a product to different suppliers/manufacturers at different locations, where the supply/production capacity for each product is often limited. Large retailers make supplier choice and hub location decisions at different stages, but small or medium retailers have small production volumes and would benefit from making both decisions simultaneously. Essentially, the agility of online retailers allows them to combine the hub location decisions with longer-term tactical supplier selection decisions. For the choice of destination, time-to-delivery has become a key competitive differentiation factor for online retailers, in addition to the traditional differentiator of price. Liu et al. (2022) showed that customer demand for online retailers is directly impacted by their logistics service quality, including shipping time, delivery time, and damaged product ratio. To increase the speed of last-mile delivery and fulfillment responsiveness, online retailers would carefully choose the destination site for a product from a set of candidate sites. These sites are all located sufficiently close to this product's primary service area and customer zone, but different sites may have different capacities and costs. To this end, we define an extension of the HLP with multiple assignments (Contreras et al. 2011a), adding the selection of origin/destination nodes (with capacity limitations) from a set of candidates for each flow. We call this extended HLP the *flow hub location problem (FHLP)*.

The FHLP is $\mathcal{NP}$-hard since a special case of FHLP (with the sets of origin/destination nodes for all flows being singletons) becomes the classical HLP with multiple assignments, which is known to be $\mathcal{NP}$-hard (Contreras and Fernández 2014). Our computational tests show that the generic branch-and-bound method of a commercial solver cannot achieve a good performance for relatively large-size FHLP instances. Additionally, although the literature has paid much attention in the past decades to finding efficient algorithms to solve various HLPs (e.g., Benders decomposition in Contreras et al. (2011a) and branch-and-price in Contreras et al. (2011c)), none of these algorithms are directly applicable to the FHLP due to its specific features. Hence, it motivates us to develop a new efficient algorithm specifically tailored for the FHLP. The proposed algorithm combines Lagrangian relaxation and Benders decomposition to exploit specific structures of the problem and solve subproblems efficiently.

The remainder of this paper is organized as follows. In Section 2, we provide a literature review on the research related to this paper. It is followed by Section 3 which presents mathematical models of the FHLP and its Benders reformulation. In Section 4, we discuss how to enhance the Benders decomposition method through the exploration of several problem properties and learning-empowered elimination tests and variable reduction. Section 5 presents computational results to show the efficiency of the proposed algorithms through a vast number of computational tests. Finally, we conclude with remarks and future research directions in Section 6. Additional computational results and model outputs are provided in the online supplement to this paper.

2. Literature Review

In this section, we review the literature relevant to the FHLPs under study and summarize the main contributions of this paper.

From the modeling perspective, this paper contributes to the literature by presenting a new extension of the classical HLP, where the selection of the origin and destination of each flow is determined simultaneously with the typical hub location decisions, i.e., the selection of intermediary hubs to lease and the resulting routing of flows. This extension is necessary to model the HLPs in e-commerce, where online sellers can dynamically change their hubs and routing as they change the suppliers or manufacturers of their products and/or the warehouses for last-mile delivery. A mixed integer linear programming (MILP) model will be formulated, and several properties of the model will be discussed to reformulate the model. To the best of our knowledge, this is the first paper studying such an extension. Several classical HLPs have been studied, such as the HLPs with single or multiple allocation (O'Kelly 1987), the p-hub covering problem (Campbell 1994), the p-hub median problem (Campbell 1996), the p-hub center problem (Ernst et al. 2009), the hub arc location problem (Campbell et al. 2005), and the cycle hub location problem (Contreras et al. 2017). More recent variants of HLPs incorporate different requirements and objectives, such as the HLPs with reliability (Mohammadi et al. 2019), competition (Gelareh et al. 2010), robustness (Wang et al. 2020, Zhao et al. 2023), multi-period (Gelareh et al. 2015), multi-objective (Monemi et al. 2021), uncertainty (Alumur et al. 2012), congestion (Elhedhli and Wu 2010), interhub link failures (Blanco et al. 2023), or economies of scale (EOS) (O'Kelly and Bryan 1998). For comprehensive reviews of the hub location literature, we refer readers to Campbell and O'Kelly (2012), Farahani et al. (2013), and Alumur et al. (2021).

The class of HLPs is known to be computationally challenging due to an exponentially large number of solutions. Because the FHLPs studied in this paper extend the classical HLPs by adding the decisions regarding the origin and destination for each flow, this class of problems can become even more challenging to solve. Further, due to the fact that e-commerce companies have more flexibility in their hub locations because of their smaller sales volumes and leasing options, the sizes of the FHLPs can be large. As such, it calls for the development of a new optimization algorithm for solving large-scale FHLPs. In general, two categories of methods have been explored to effectively solve the HLPs, namely, heuristic methods and exact methods. Heuristics aim to find near-optimal solutions within a reasonable amount of computational time and thus have been developed to solve large-scale optimization problems. Popular heuristics for the HLPs include genetic algorithms (Meng and Wang 2011), tabu search (Bütün et al. 2021), adaptive neighborhood search (Wu et al. 2022c), simulated annealing (Ghaffarinasab et al. 2018), particle swarm optimization (Özğun-Kibiroğlu et al. 2019, Maiyar and Thakkar 2019), as well as their variants and combinations, such as the hybrid tabu search and greedy random adaptive search algorithm for the HLP with EOS in Klincewicz (2002). More recently, learning-based algorithms have attracted much attention and have been successfully applied to network design problems, such as the learning-based probabilistic tabu search in Guan et al. (2018) and unsupervised learning-driven matheuristic in Wu et al. (2022b).

Despite the effectiveness of heuristics in finding good solutions in a relatively short amount of time, they typically lack guarantees on the solution quality. Due to the tactical nature of the HLPs and their

long-term impacts, companies are often willing to spend more computational time in exchange for a better solution quality. As such, exact methods have been developed for solving HLPs with proven optimal solutions or theoretical bounds. These exact methods are typically based on the mathematical programming framework. For example, exact methods based on branch-and-bound have been proposed for solving uncapacitated multiple allocation HLPs (Klincewicz 1996, Mayer and Wagner 2002), single and multiple allocation p-hub median problems (Ernst and Krishnamoorthy 1998), and the HLP with profits (Armaghan et al. 2018); methods based on branch-and-price have been proposed for the HLPs with single assignment, where Lagrangian relaxation is used to obtain lower bounds on the restricted master problem in Contreras et al. (2011c), and several facet-defining inequalities and traffic variables are used to enhance the efficiency of the algorithm in Labbé et al. (2005); a branch-and-cut based algorithm is developed in Contreras et al. (2017) for the cycle hub location problem, which uses a flow-based formulation and two families of mixed-dicut inequalities as a lower bounding procedure at nodes of the enumeration tree. Another popular class of exact methods in solving the HLPs is Benders decomposition, which decomposes a large-size problem into smaller solvable subproblems and a master problem. For example, Benders decomposition algorithms have been developed to solve the uncapacitated HLP with multiple assignments in telecommunication and transportation systems (de Camargo et al. 2008), the multiple allocation HLPs with EOS (de Camargo et al. 2009), the tree of hub location problems (de Sá et al. 2013), and the uncapacitated multiple allocation HLPs with EOS and node congestion (Waleed and Diabat 2020). It is worth mentioning that variable fixing and reduction, approximate Pareto-optimal cuts, and acceleration techniques can be used to efficiently solve master problems (Contreras et al. 2011a, 2012, Taherkhani et al. 2020). A recent Benders decomposition implementation proposed by Taherkhani et al. (2020) can optimally solve the profit-maximizing HLP with up to 500 nodes and three demand classes.

From the solution methodology perspective, this paper contributes to the algorithmic development of the hub location literature by effectively combining the Benders decomposition with machine learning and the Lagrangian relaxation, as well as a few other algorithmic enhancements. While Lagrangian relaxation-based methods have been developed to solve several HLPs with the considerations of capacities, congestion, or competitive environment (Elhedhli and Wu 2010, Gelareh et al. 2010, He et al. 2015, Wu et al. 2021), in this paper, a Lagrangian relaxation heuristic is developed to generate initial solutions for the FHLP, which are then fed into the Benders decomposition method to speed up the search. Further, the standard Benders decomposition algorithm is enhanced through the exploration of several problem properties, such as a clustering-empowered multi-commodity Benders reformulation, learning-empowered elimination tests and variable reduction, and the generation of strong Pareto-optimal cuts. We have created a large number of test instances, and the computational results indicate that the proposed method can obtain optimal solutions in regular-size test instances and achieve much better solution qualities in large-size test instances than benchmark methods, such as the Cplex solver and the elimination test (Contreras et al. 2011a, Taherkhani et al. 2020).

While this paper focuses on solving deterministic FHLPs, it is worth mentioning that some recent research has explored the impact of uncertainties in HLPs. Examples of such uncertainties include demand

uncertainty (Correia et al. 2018, Rostami et al. 2021), random disruptions (Torkestani et al. 2018, Mohammadi et al. 2019, Shen et al. 2021), coexistence of stochastic demand and uncertain revenue (Taherkhani et al. 2021), and risk aversion (Kargar and Mahmutoğulları 2022). A future extension of the FHLP under study is to consider uncertainties brought by different levels of reliability of suppliers (origins) and different levels of demand uncertainty of customer zones (destinations). While previous research has demonstrated the effectiveness of using fast machine learning techniques to predict solution values of the second-stage subproblems, rather than solving them exactly with costly computations, in expediting the solution process of Benders decomposition for stochastic models (Larsen et al. 2023), we advance the research literature by using machine learning techniques to predict the likelihood of a hub being closed at an optimal solution point, and then use the insight to guide elimination tests and a variable reduction procedure to expedite the solution process of Benders decomposition.

3. Mathematical Models

This section provides a MILP formulation for the problem under study. Figure 1 illustrates the location choice and routing decisions in a three-hub one-commodity FHLP, with a comparison to the traditional HLP. Here, o, d, and h in the figure represent the origin, destination, and hub node, respectively. For the traditional HLP, given an origin o_1 and a destination d_1, the problem determines a transportation route from the set of hub nodes $\{h_1, h_2, h_3\}$, as shown in Figure 1(a). In comparison, the FHLP determines not only the hubs but also the origin selection from the set $\{o_1, o_2\}$ and the destination from the set $\{d_1, d_2\}$. Figure 1(b) shows four out of nine possible solutions in the FHLP given h_1 and h_2 as hubs used, expanding the solution space of the traditional HLP.

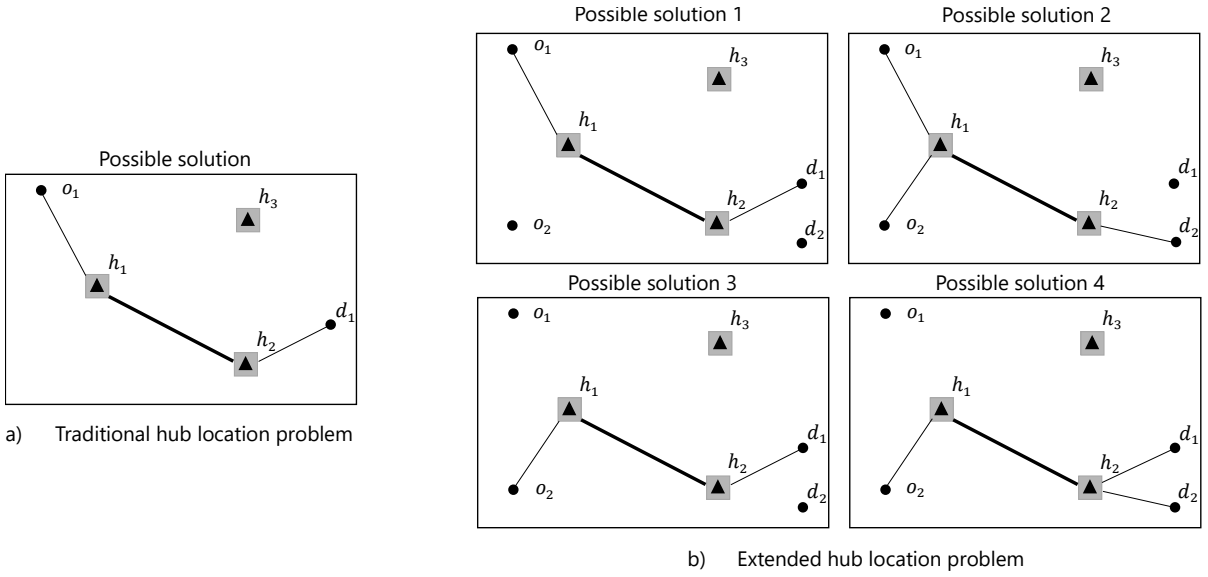

Figure 1 **HLP vs. FHLP**

In this paper, we assume that the e-commerce company aims to determine origin/destination nodes for each commodity (i.e., product) from the set of alternative origins/destinations. The assignment of the

origin/destination nodes incurs some associated costs. The objective of the FHLP is to minimize the total transshipment cost, hub location cost, and assignment cost on the origin and destination nodes.

3.1. The Problem and MILP Model

Let $G = (N, A)$ be a directed network, where N is the set of nodes and A is the set of arcs with $(h_1, h_2) \in A$ connecting nodes h_1 and h_2. Let $H \subseteq N$ denote the set of hub locations. The FHLP considers a set K of commodities indexed by k ($k \in K$) whose demand w_k is known. The origin o and destination d for each commodity k are not predetermined and need to be selected from a set ($O_k \subseteq N$) of origin nodes and a set ($D_k \subseteq N$) of destination nodes, respectively. The total demand of commodities assigned to each origin (o) and destination (d) node cannot violate their respective capacities, denoted by c_{ok} and c_{dk} for each commodity k. We assume that the combined capacities at origins and destinations are sufficient for satisfying the demand, i.e., $\sum_{o \in O_k} c_{ok} \geq w_k$ and $\sum_{d \in D_k} c_{dk} \geq w_k$, $\forall\ k \in K$. Every commodity ($k \in K$) must be assigned at least one origin and one destination, but multiple origins and multiple destinations can be assigned to the same commodity. Assigning an origin o (destination d) to a commodity k incurs an assignment cost of p_{ok} (p_{dk}). Given that hubs are uncapacitated, hub nodes are fully interconnected, and distances follow the triangle inequality, every path between an origin and a destination contains at least one and at most two hubs. Herein, for each commodity k, a path between two nodes takes the form (o, h_1, h_2, d), where $h_1, h_2 \in H$. Further, let Δ_{h_1,h_2} represent the travel distance from node h_1 to node h_2, where $\Delta_{h_1 h_2}$ is assumed to be symmetric. Note that when $h_1 = h_2$, the path only contains one hub with $\Delta_{h_1 h_2} = 0$.

The transportation cost $t_{odh_1h_2k}$ for moving the whole demand of commodity k along the path (o, h_1, h_2, d) is proportional to $w_k \cdot (\sigma_1 \cdot \Delta_{oh_1} + \sigma_2 \cdot \Delta_{h_1h_2} + \sigma_3 \cdot \Delta_{h_2d})$, where $\sigma_2 < \sigma_1$ and $\sigma_2 < \sigma_3$ due to EOS for the transportation between hubs (Contreras et al. 2011a). For each $h \in H$, f_h is the fixed cost for leasing hub h. The overall objective is to minimize the total costs, including the hub-leasing costs (f_h), transportation costs ($t_{odh_1h_2k}$), and assignment costs (p_{ok} and p_{dk}). We summarize all the notation and provide the model formulation as follows.

Sets:

O Set of origin nodes for all commodities, indexed by o.
D Set of destination nodes for all commodities, indexed by d.
H Set of potential hubs, indexed by h, h_1, and h_2.
N Set of all nodes, where $N = O \cup D \cup H$.
A Set of arcs.
K Set of commodities, indexed by k.
O_k Set of nodes that can serve as the origin for commodity k, where $O_k \subseteq O$.
D_k Set of nodes that can serve as the destination for commodity k, where $D_k \subseteq D$.
K_o Set of commodities that can originate from node o, where $K_o \subseteq K$.
K_d Set of commodities that can be delivered to node d, where $K_d \subseteq K$.

Parameters:

w_k Demand of commodity k.
$t_{odh_1h_2k}$ Transportation cost for delivering the total demand (w_k) of commodity k from origin o to destination d through hubs h_1 and h_2.
f_h Leasing cost of hub h.
p_{ok} Assignment cost for selecting node o as an origin for commodity k.
p_{dk} Assignment cost for selecting node d as a destination for commodity k.
c_{ok} Capacity of origin node o for commodity k.
c_{dk} Capacity of destination node d for commodity k.

Variables:

$x_{odh_1h_2k}$ Transportation variables representing the percentage of commodity k transported from origin o to destination d through hubs h_1 and h_2.
z_h Binary hub-leasing variables; $z_h = 1$ if hub h is leased, and 0 otherwise.
y_{ok} Binary commodity origin assignment variables; $y_{ok} = 1$ if node o is selected as an origin for commodity k, and 0 otherwise.
u_{dk} Binary commodity destination assignment variables; $u_{dk} = 1$ if node d is selected as a destination for commodity k, and 0 otherwise.

Using the above notation, the FHLP can be formulated as the following MILP:

$$\min \sum_{h\in H} f_h \cdot z_h + \sum_{k\in K}\sum_{o\in O_k}\sum_{d\in D_k}\sum_{h_1\in H}\sum_{h_2\in H} t_{odh_1h_2k}\cdot x_{odh_1h_2k} + \sum_{k\in K}\sum_{o\in O_k} p_{ok}\cdot y_{ok} + \sum_{k\in K}\sum_{d\in D_k} p_{dk}\cdot u_{dk} \tag{1}$$

Subject to:

$$\sum_{o\in O_k}\sum_{d\in D_k}\sum_{h_1\in H}\sum_{h_2\in H} x_{odh_1h_2k} = 1, \quad \forall\, k\in K, \tag{2}$$

$$\sum_{o\in O_k}\sum_{d\in D_k}\sum_{h_1\in H} x_{odhh_1k} + \sum_{o\in O_k}\sum_{d\in D_k}\sum_{h_1\in H\setminus\{h\}} x_{odh_1hk} \le z_h, \quad \forall\, k\in K,\ h\in H, \tag{3}$$

$$\sum_{d\in D_k}\sum_{h_1\in H}\sum_{h_2\in H} w_k\cdot x_{odh_1h_2k} \le c_{ok}\cdot y_{ok}, \quad \forall\, k\in K,\ o\in O_k, \tag{4}$$

$$\sum_{o\in O_k}\sum_{h_1\in H}\sum_{h_2\in H} w_k\cdot x_{odh_1h_2k} \le c_{dk}\cdot u_{dk}, \quad \forall\, k\in K,\ d\in D_k, \tag{5}$$

$$y_{ok}\in\{0,1\},\ u_{dk}\in\{0,1\}, \quad \forall\, k\in K,\ o\in O_k,\ d\in D_k, \tag{6}$$

$$x_{odh_1h_2k}\ge 0, \quad \forall\, k\in K,\ o\in O_k,\ d\in D_k,\ h_1,h_2\in H, \tag{7}$$

$$z_h\in\{0,1\}, \quad \forall\, h\in H. \tag{8}$$

In this formulation, the objective function (1) is to minimize the total hub-leasing, transportation, and assignment costs. Constraints (2) ensure that the demand for every commodity is fully satisfied. Constraints (3) ensure that a hub must be leased if any commodity goes through the hub. Constraints (4) ensure that the total demand of each commodity originating from a node does not exceed its capacity. Constraints (5) enforce that the total demand of each commodity delivered to a node does not exceed its capacity. Constraints (4) and (5) also ensure that a commodity can go from an origin (to a destination) only if this origin (destination) is selected for this commodity. Constraints (6)–(7) enforce integrality and non-negativity requirements for the different types of variables.

3.2. Variable Reduction

We notice that some properties can be utilized to preprocess the above formulation so as to reduce the number of variables $x_{odh_1h_2k}$. Some of these properties are inherited from the uncapacitated hub location problem discussed in Contreras et al. (2011a). Specifically, define a set of hub edges $e \in E$, where E is the set of all subsets of H containing one or two hubs, and denote e as $\{e_1\}$ if $|e| = 1$ and as $\{e_1, e_2\}$ if $|e| = 2$. Then, about half of the $x_{odh_1h_2k}$ variables can be ruled out by simply using an undirected transportation cost for every hub edge (Hamacher et al. 2004). In an optimal solution, every commodity needs at most one direction of a hub edge, i.e., the one with the lowest shipping cost. The undirected shipping cost t_{odek} for each $e \in E$, $k \in K$ is defined as $t_{odek} = \min\{t_{odh_1h_2k}, t_{odh_2h_1k}\}$ if $e = \{h_1, h_2\}$, and $t_{odek} = t_{odh_1h_1k}$ if $e = \{h_1\}$. Furthermore, it can be verified that in any optimal solution, no commodities will be transported through a hub edge containing two different hubs whenever it is cheaper to transport it through only one of them (Contreras et al. 2011a).

Property 3.1 *For every $k \in K$, $o \in O_k$, $d \in D_k$, and $e = \{e_1, e_2\} \in E$, if $t_{odek} > \min\left\{t_{od\{e_1\}k}, t_{od\{e_2\}k}\right\}$, then $x_{odek} = 0$ in any optimal solution of the FHLP.*

Utilizing this property can lead to a more compact formulation with fewer variables. To this end, we define a set of candidate hub edges for each commodity $k \in K$, $o \in O_k$, $d \in D_k$ as

$$E_{odk} = \Big\{ \{e \in E : |e| = 1\} \cup \{e \in E : |e| = 2 \text{ and } (t_{odek} < \min\{t_{od\{e_1\}k}, t_{od\{e_2\}k}\})\}.$$

The mathematical model can be therefore reformulated to problem $\mathcal{P}$ as follows:

$$(\mathcal{P}): \quad \min \sum_{h\in H} f_h \cdot z_h + \sum_{k\in K}\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} t_{odek}\cdot x_{odek} + \sum_{k\in K}\sum_{o\in O_k} p_{ok}\cdot y_{ok} + \sum_{k\in K}\sum_{d\in D_k} p_{dk}\cdot u_{dk} \tag{9}$$

Subject to:

$$\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek} = 1, \quad \forall\, k \in K, \tag{10}$$

$$\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}:h\in e} x_{odek} \le z_h, \quad \forall\, k \in K,\ h \in H, \tag{11}$$

$$\sum_{d\in D_k}\sum_{e\in E_{odk}} w_k \cdot x_{odek} \le c_{ok}\cdot y_{ok}, \quad \forall\, k \in K,\ o \in O_k, \tag{12}$$

$$\sum_{o\in O_k}\sum_{e\in E_{odk}} w_k \cdot x_{odek} \le c_{dk}\cdot u_{dk}, \quad \forall\, k \in K,\ d \in D_k, \tag{13}$$

$$y_{ok} \in \{0,1\},\ u_{dk} \in \{0,1\}, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k, \tag{14}$$

$$x_{odek} \ge 0, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k,\ e \in E_{odk}, \tag{15}$$

$$z_h \in \{0,1\}, \quad \forall\, h \in H. \tag{16}$$

3.3. Benders Decomposition Formulation

The presence of a large number of variables in the MILP formulation (9)–(16), particularly those binary variables, can make the problem challenging to solve using a standard MILP solver. The structure of the formulation makes it suitable to apply Benders decomposition to separate the original problem into smaller problems that are individually easier to solve. Specifically, a master problem is solved over a first subset of variables of the original problem, given an estimate for the cost of the remaining second subset of variables. Then, a subproblem is solved over the second subset of variables, given the solution of the first subset of variables obtained in the master problem. If the subproblem is infeasible or its cost exceeds the current estimate, then so-called Benders cuts are generated and added to the master problem. This process repeats until the optimality or infeasibility of the problem is proved. This subsection provides the Benders reformulation and the solution framework based on the Benders formulation.

In the Benders reformulation of the FHLP, the hub-leasing and origin/destination assignment variables (z, y, u) are handled in the master problem and the transportation variables are handled in the subproblems. Specifically, let $G_1 = \mathbb{B}^{|H|}$, $G_2 = \mathbb{B}^{\sum_{k\in K}|O_k|}$, and $G_3 = \mathbb{B}^{\sum_{k\in K}|D_k|}$ denote the sets of binary vectors associated with z_h, y_{ok}, and u_{dk}, respectively. For any fixed vectors $\tilde{z} \in G_1$, $\tilde{y} \in G_2$, and $\tilde{u} \in G_3$, the primal subproblem $(\mathcal{PS})$ over the variables x_{odek} is given as follows:

$$(\mathcal{PS}): \quad \min \left\{ \sum_{k\in K} \sum_{o\in O_k} \sum_{d\in D_k} \sum_{e\in E_{odk}} t_{odek} \cdot x_{odek} \right\} \tag{17}$$

subject to:

$$\sum_{o\in O_k} \sum_{d\in D_k} \sum_{e\in E_{odk}} x_{odek} = 1, \quad \forall\, k \in K, \tag{18}$$

$$\sum_{o\in O_k} \sum_{d\in D_k} \sum_{e\in E_{odk}: h\in e} x_{odek} \le \tilde{z}_h, \quad \forall\, k \in K,\ h \in H, \tag{19}$$

$$\sum_{d\in D_k} \sum_{e\in E_{odk}} w_k \cdot x_{odek} \le c_{ok} \cdot \tilde{y}_{ok}, \quad \forall\, k \in K,\ o \in O_k, \tag{20}$$

$$\sum_{o\in O_k} \sum_{e\in E_{odk}} w_k \cdot x_{odek} \le c_{dk} \cdot \tilde{u}_{dk}, \quad \forall\, k \in K,\ d \in D_k, \tag{21}$$

$$x_{odek} \ge 0, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k,\ e \in E_{odk}. \tag{22}$$

While the master problem provides a lower bound on the original problem, the subproblem is used to provide an upper bound. Thus, $\mathcal{PS}$ is reformulated using its dual representation to form the dual subproblem. By associating the dual variables α_k, θ_{hk}, γ_{ok}, and τ_{dk} to constraints (18)–(21), respectively, the dual subproblem $\mathcal{DS}$ is given as follows:

$$(\mathcal{DS}): \quad \max \left\{ \sum_{k\in K} \alpha_k - \sum_{k\in K} \sum_{h\in H} \tilde{z}_h \cdot \theta_{hk} - \sum_{k\in K} \sum_{o\in O_k} c_{ok} \cdot \tilde{y}_{ok} \cdot \gamma_{ok} - \sum_{k\in K} \sum_{d\in D_k} c_{dk} \cdot \tilde{u}_{dk} \cdot \tau_{dk} \right\} \tag{23}$$

subject to:

$$\alpha_k - \theta_{e_1 k} - \theta_{e_2 k} - w_k \cdot \gamma_{ok} - w_k \cdot \tau_{dk} \le t_{odek}, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k,\ e \in E_{odk},\ |e| = 2, \tag{24}$$

$$\alpha_k - \theta_{e_1 k} - w_k \cdot \gamma_{ok} - w_k \cdot \tau_{dk} \le t_{odek}, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k,\ e \in E_{odk},\ |e| = 1, \tag{25}$$

$$\theta_{hk} \geq 0, \quad \forall\, k \in K,\ h \in H, \tag{26}$$

$$\gamma_{ok} \geq 0,\ \tau_{dk} \geq 0, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k. \tag{27}$$

Note that both $\mathcal{PS}$ and $\mathcal{DS}$ can be decomposed to $|K|$ independent subproblems $\mathcal{PS}_k$ and $\mathcal{DS}_k$, respectively, given any $k \in K$. After solving $\mathcal{DS}$, there are three possible outcomes. If $\mathcal{DS}$ has a finite optimal solution, and so does $\mathcal{PS}$ by the duality theory, the upper bound can be updated to the current optimal value of the subproblem plus its corresponding hub-leasing and assignment costs if it is smaller than the current upper bound, and a new constraint is added to the master problem by considering this solution; if $\mathcal{DS}$ is unbounded above and thus $\mathcal{PS}$ is infeasible, a constraint is added to the master problem to remove this solution; and if $\mathcal{DS}$ is infeasible and thus $\mathcal{PS}$ is infeasible or unbounded below, the procedure terminates. The following property can be used to strengthen the master problem.

Proposition 3.2 *For any vectors $\tilde{z} \in G_1$, $\tilde{y} \in G_2$, and $\tilde{u} \in G_3$ satisfying $\sum_{h\in H} \tilde{z}_h \geq 1$, $\sum_{o\in O_k} \tilde{y}_{ok} \geq 1$ for all $k \in K$, and $\sum_{d\in D_k} \tilde{u}_{dk} \geq 1$ for all $k \in K$, the primal subproblem $\mathcal{PS}$ is either infeasible or feasible and bounded, and the dual subproblem $\mathcal{DS}$ is either unbounded or feasible and bounded.*

Proof. We can prove this proposition by verifying the validity of the following three points. 1) The infeasibility of the primal subproblem is possible, 2) the feasibility of the primal subproblem is possible, and 3) the primal subproblem cannot be unbounded. For point 1), the infeasibility of the primal subproblem is possible because, for a given commodity, the combined capacity of the selected origins (destinations) may be insufficient for its demand. For point 2), the feasibility of the primal subproblem is possible because $\mathcal{P}$ has a finite optimal solution. For point 3), from constraints (18)–(22) and the fact that the transportation costs t_{odek} are finite, any feasible solution of the primal subproblem must be bounded. Hence, the primal subproblem is feasible and bounded or infeasible. According to strong duality, the dual subproblem is either feasible and bounded or unbounded. □

Let $\mathbb{P}$ denote the polyhedron defined by constraints (24)–(27), and let $\mathbb{P}^e$ and $\mathbb{P}^u$ represent the set of extreme points and the set of unbounded rays of $\mathbb{P}$, respectively. Further, let ζ denote the total transportation cost obtained by satisfying the demand. The Benders master problem ($\mathcal{MP}$) can then be formulated as follows:

$$(\mathcal{MP}):\ \min \left\{ \sum_{h\in H} f_h \cdot z_h + \sum_{k\in K} \sum_{o\in O_k} p_{ok} \cdot y_{ok} + \sum_{k\in K} \sum_{d\in D_k} p_{dk} \cdot u_{dk} + \zeta \right\} \tag{28}$$

subject to:

$$\zeta \geq \sum_{k\in K} \alpha_k - \sum_{k\in K} \sum_{h\in H} z_h \cdot \theta_{hk} - \sum_{k\in K} \left(\sum_{o\in O_k} c_{ok} \cdot y_{ok} \cdot \gamma_{ok} - \sum_{d\in D_k} c_{dk} \cdot u_{dk} \cdot \tau_{dk} \right),\ \forall\ (\alpha, \theta, \gamma, \tau) \in \mathbb{P}^e, \tag{29}$$

$$\sum_{k\in K} \alpha_k - \sum_{k\in K} \sum_{h\in H} z_h \cdot \theta_{hk} - \sum_{k\in K} \left(\sum_{o\in O_k} c_{ok} \cdot y_{ok} \cdot \gamma_{ok} - \sum_{d\in D_k} c_{dk} \cdot u_{dk} \cdot \tau_{dk} \right) \leq 0,\ \forall\ (\alpha, \theta, \gamma, \tau) \in \mathbb{P}^u, \tag{30}$$

$$\sum_{o \in O_k} y_{ok} \geq 1, \quad \forall\, k \in K, \tag{31}$$

$$\sum_{d \in D_k} u_{dk} \geq 1, \quad \forall\, k \in K, \tag{32}$$

$$\sum_{h \in H} z_h \geq 1, \tag{33}$$

$$y_{ok} \in \{0,1\},\ u_{dk} \in \{0,1\}, \quad \forall\, k \in K,\ o \in O_k,\ d \in D_k, \tag{34}$$

$$z_h \in \{0,1\}, \quad \forall\, h \in H. \tag{35}$$

Here, constraints (29) are cuts added when $\mathcal{DS}$ is feasible and bounded, and constraints (30) are cuts generated when $\mathcal{DS}$ is unbounded. Inequalities (31)–(32) impose that for each commodity, at least one origin and one destination must be selected. Constraint (33) imposes that at least one hub must be selected. These constraints are added to improve the quality of the initial Benders cuts. We note that the master problem $\mathcal{MP}$ contains an exponential number of constraints. A cutting plane approach can be employed to solve $\mathcal{MP}$. We can iteratively solve relaxed master problems containing a small subset of constraints (29) and (30) associated with the extreme points of $\mathbb{P}^e$ and the unbounded rays of $\mathbb{P}^u$, respectively. The feasible solution provides some substantiated values $(\tilde{z}, \tilde{y}, \tilde{u})$ to the dual subproblem $\mathcal{DS}$. The dual subproblem is then solved to optimality or with unbounded rays, and its solution is an extreme point of $\mathbb{P}^e$ (or an unbounded ray of $\mathbb{P}^u$), which can be used to generate a new constraint of (29) (or (30)) for the relaxed master problem. Afterward, we solve the relaxed master problem to obtain new solution values $(\tilde{z}, \tilde{y}, \tilde{u})$. This procedure is iteratively executed until an optimal solution to the original FHLP is found.

A pseudocode of the Benders decomposition algorithm is presented in Algorithm 1. In this algorithm, we let $\mathcal{LB}$ and $\mathcal{UB}$ represent a lower bound and an upper bound on the optimal objective value of the FHLP, respectively. Let ε denote the iteration count, and let $\mathbb{P}^e_\varepsilon$ and $\mathbb{P}^u_\varepsilon$ be the restricted set of extreme points and the restricted set of unbounded rays of $\mathbb{P}$ at iteration ε, respectively. Additionally, we let $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ be an optimal solution vector of $\mathbb{P}^e_\varepsilon$ and let $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ denote the dual subproblem given $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$. Let $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon)$ be the relaxed master problem in which $\mathbb{P}^e$ and $\mathbb{P}^u$ are replaced by $\mathbb{P}^e_\varepsilon$ and $\mathbb{P}^u_\varepsilon$ in $\mathcal{MP}$, respectively. Moreover, we let $z(\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon))$ and $z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon))$ be the optimal objective values of the dual subproblem $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ and the master problem $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon)$, respectively.

The computational efficiency of Algorithm 1 generally depends on: (i) the computational effort required to solve $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon)$, (ii) the computational effort required to solve $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$, and (iii) the number of iterations before termination (convergence). We will, therefore, present several techniques to enhance the performance of Algorithm 1 by tackling these points.

3.4. Feasibility of Primal Subproblems

From Proposition 3.2, the primal subproblem can be infeasible and the dual subprogram can be unbounded. Rather than solving the dual subproblems to check if they are unbounded or not, we can use the rules in Propositions 3.3 to check the feasibility. If unbounded, we can directly solve the modified dual subproblems to obtain the unbounded ray. Performing this step at iterations of the algorithm could save computation time.

Algorithm 1: The Benders Decomposition Algorithm

$\mathcal{LB} \leftarrow -\infty$, $\mathcal{UB} \leftarrow +\infty$, $\varepsilon \leftarrow 1$, $\mathbb{P}^e_\varepsilon \leftarrow \emptyset$, $\mathbb{P}^u_\varepsilon \leftarrow \emptyset$;

Obtain the initial solution of $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$;

repeat

  Solve $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$;

  if *Unbounded* **then**

    Obtain unbounded ray $(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon) \in \mathbb{P}^u$;

    $\mathbb{P}^u_{\varepsilon+1} \leftarrow \mathbb{P}^u_\varepsilon \cup \{(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon)\}$;

  else

    Obtain extreme point $(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon) \in \mathbb{P}^e$ and $z(\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon))$;

    $\mathcal{UB} \leftarrow \min\{\mathcal{UB},$

      $z(\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)) + \sum_{h\in H} f_h \cdot \tilde{z}_h + \sum_{k\in K}\sum_{o\in O_k} p_{ok} \cdot \tilde{y}_{ok} + \sum_{k\in K}\sum_{d\in D_k} p_{dk} \cdot \tilde{u}_{dk}\}$;

    $\mathbb{P}^e_{\varepsilon+1} \leftarrow \mathbb{P}^e_\varepsilon \cup \{(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon)\}$;

  end

  $\varepsilon \leftarrow \varepsilon + 1$;

  Solve $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon)$ to obtain $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ and $z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon))$;

  $\mathcal{LB} \leftarrow z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon))$;

until $\mathcal{UB} = \mathcal{LB}$;

Proposition 3.3 *For any $k \in K$, let $\tilde{O}_k = \{o \in O_k | \tilde{y}_{ok} = 1\}$ and $\tilde{D}_k = \{d \in D_k | \tilde{u}_{dk} = 1\}$. The primal subproblem $\mathcal{PS}$ is infeasible if $w_k > \sum_{o\in\tilde{O}_k} c_{ok}$ or $w_k > \sum_{d\in\tilde{D}_k} c_{dk}$ for any $k \in K$.*

Proof. For any $k \in K$, sum both sides of Eq. (20) over $o \in O_k$, we have $\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} w_k \cdot x_{odek} \le \sum_{o\in O_k} c_{ok} \cdot \tilde{y}_{ok}$. From Eq. (18), $\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} w_k \cdot x_{odek} = w_k$, and $\sum_{o\in O_k} c_{ok} \cdot \tilde{y}_{ok} = \sum_{o\in\tilde{O}_k} c_{ok}$. Therefore, if $w_k > \sum_{o\in\tilde{O}_k} c_{ok}$ for any $k \in K$, $\mathcal{PS}$ is infeasible. In a similar vein, we can show that, if $w_k > \sum_{d\in\tilde{D}_k} c_{dk}$ for any $k \in K$, $\mathcal{PS}$ is also infeasible. □

4. Algorithmic Enhancements

In this section, we present several ways of improving the convergence and stability of the Benders decomposition algorithm described in Algorithm 1.

4.1. Lagrangian Relaxation Heuristic for Generating Initial Solution of $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$

The quality of the initial solution can significantly affect the efficiency and effectiveness of the Benders decomposition algorithm. Specifically, a good initial solution can help the algorithm converge faster and find a better quality solution, while a poor initial solution may cause the algorithm to converge slowly or get stuck in a local optimum. Therefore, it is important to provide a good initial solution of $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ in Algorithm 1. In this paper, we develop a Lagrangian relaxation heuristic to obtain an initial solution.

4.1.1. Lagrangian Formulation and Solution. Observing the FHLP model in (9)–(16), the binary variables in constraints (11)–(13) pose challenges in solving the model, and thus, the corresponding constraints will be relaxed. Let ψ_{kh}, ξ_{ok}, and ξ_{dk} denote the vectors of Lagrange multipliers for constraints

(11)–(13), respectively. After relaxing constraints (11)–(13) and dualizing their terms into the objective function (9), the Lagrangian subproblem $L(\psi,\xi)$ is given as follows:

$$(L(\psi,\xi)): \quad \min \sum_{h\in H} \bar{f}_h \cdot z_h + \sum_{k\in K}\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} \bar{t}_{odek}\cdot x_{odek} + \sum_{k\in K}\sum_{o\in O_k} \bar{p}_{ok}\cdot y_{ok} + \sum_{k\in K}\sum_{d\in D_k} \bar{p}_{dk}\cdot u_{dk}$$

Subject to:

(10), (14)–(16),

where $\bar{f}_h = f_h - \sum_{k\in K}\psi_{kh}$, $\bar{p}_{ok} = p_{ok} - c_{ok}\cdot\xi_{ok}$, $\bar{p}_{dk} = p_{dk} - c_{dk}\cdot\xi_{dk}$, and

$$\bar{t}_{odek} = \begin{cases} t_{odek} + \psi_{ke_1} + \psi_{ke_2} + w_k\cdot(\xi_{ok}+\xi_{dk}), & \forall\ k\in K,\ o\in O_k,\ d\in D_k,\ e\in E_{odk},\ |e|=2,\\ t_{odek} + \psi_{ke_1} + w_k\cdot(\xi_{ok}+\xi_{dk}), & \forall\ k\in K,\ o\in O_k,\ d\in D_k,\ e\in E_{odk},\ |e|=1. \end{cases}$$

Since constraints (12) and (13) are relaxed, no constraints in $L(\psi,\xi)$ connect variables x to (y,u), potentially leading to low-quality or infeasible solutions. Hence, we add two sets of constraints, (36) and (37), to $L(\psi,\xi)$, which do not significantly increase the complexity of the Lagrangian problem.

Proposition 4.1 *The following constraints* (36) *and* (37) *are valid for problem* $\mathcal{P}$.

$$\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek} \le y_{ok}, \quad \forall\ k\in K,\ o\in O_k, \tag{36}$$

$$\sum_{o\in O_k}\sum_{e\in E_{odk}} x_{odek} \le u_{dk}, \quad \forall\ k\in K,\ d\in D_k. \tag{37}$$

Proof. For any $k\in K$ and $o\in O_k$, constraint (36) is equivalent to $\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek}\le 0$ if $y_{ok}=0$ and $\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek}\le 1$ if $y_{ok}=1$. When $y_{ok}=0$, from Eq. (12), $x_{odek}=0$ for all $d\in D_k$ and $e\in E_{odk}$, and thus $\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek}\le 0$ is valid; when $y_{ok}=1$, from Eq. (10), $\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek}\le 1$ is valid. Therefore, Eq. (36) is a valid constraint for problem $\mathcal{P}$. In a similar vein, we can show that Eq. (37) is also a valid constraint for problem $\mathcal{P}$. □

Noticing that the hub-leasing variables z are disjoint from the other variables (x,y,u) in the Lagrangian subproblem $L(\psi,\xi)$, it can be decomposed to two smaller subproblems with one over the hub-leasing variables and the other over the remaining decision variables. We define these two subproblems as $L_1(\psi)$ and $L_2(\psi,\xi)$, given as follows:

$$(L_1(\psi)): \quad \min\left\{\sum_{h\in H}\bar{f}_h\cdot z_h \,|\, z_h\in\{0,1\},\ \forall\ h\in H\right\},$$

and

$$(L_2(\psi,\xi)): \quad \min \sum_{k\in K}\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} \bar{t}_{odek}\cdot x_{odek} + \sum_{k\in K}\sum_{o\in O_k} \bar{p}_{ok}\cdot y_{ok} + \sum_{k\in K}\sum_{d\in D_k} \bar{p}_{dk}\cdot u_{dk}$$

subject to:

$(10), (14)-(15), (36)-(37).$

Subproblem $L_2(\psi,\xi)$ is a semi-assignment problem that can be further decomposed into $|K|$ independent semi-assignment problems, corresponding to each commodity $k \in K$, with the formulation for any given k provided as follows:

$$(L_2^k(\psi,\xi)): \quad \min \sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} \bar{t}_{odek}\cdot x_{odek} + \sum_{o\in O_k}\bar{p}_{ok}\cdot y_{ok} + \sum_{d\in D_k}\bar{p}_{dk}\cdot u_{dk} \tag{38}$$

subject to:

$$\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek} = 1, \tag{39}$$

$$\sum_{d\in D_k}\sum_{e\in E_{odk}} x_{odek} \le y_{ok}, \quad \forall\, o\in O_k, \tag{40}$$

$$\sum_{o\in O_k}\sum_{e\in E_{odk}} x_{odek} \le u_{dk}, \quad \forall\, d\in D_k, \tag{41}$$

$$y_{ok}\in\{0,1\},\ u_{dk}\in\{0,1\}, \quad \forall\, o\in O_k,\ d\in D_k, \tag{42}$$

$$x_{odek}\ge 0,\ \forall\, o\in O_k,\ d\in D_k,\ e\in E_{odk}. \tag{43}$$

The optimal solutions and the optimal objective values of $L_1(\psi)$, $L_2^k(\psi,\xi)$, and $L(\psi,\xi)$ can be conveniently computed as specified by Proposition 4.2.

Proposition 4.2 *The optimal solution of subproblem $L_1(\psi)$ and its optimal objective value $\hat{L}_1(\psi)$ are given by*

$$z_h^* = \begin{cases} 1, & \text{if } \bar{f}_h < 0 \\ 0, & \text{otherwise} \end{cases}, \quad \forall h\in H, \tag{44}$$

$$\hat{L}_1(\psi) = \sum_{h\in H}\min(0,\bar{f}_h). \tag{45}$$

The optimal solution of subproblem $L_2^k(\psi,\xi)$ for any $k\in K$ and its optimal objective value $\hat{L}_2^k(\psi,\xi)$ are given by

$$x_{odek}^* = \begin{cases} 1, & \text{if } (o,d,e) = (o_k^*, d_k^*, e_k^*) \\ 0, & \text{otherwise} \end{cases}, \quad \forall o\in O_k, d\in D_k, e\in E_{odk}, \tag{46}$$

$$y_{ok}^* = \begin{cases} 1, & \text{if } \bar{p}_{ok}\le 0 \text{ or } o = o_k^* \\ 0, & \text{otherwise} \end{cases}, \quad \forall o\in O_k, \tag{47}$$

$$u_{dk}^* = \begin{cases} 1, & \text{if } \bar{p}_{dk}\le 0 \text{ or } d = d_k^* \\ 0, & \text{otherwise} \end{cases}, \quad \forall d\in D_k, \tag{48}$$

$$\hat{L}_2^k(\psi,\xi) = \min_{o\in O_k, d\in D_k, e\in E_{odk}} \bar{\bar{t}}_{odek} + \sum_{o\in O_k}\bar{p}_{ok}^- + \sum_{d\in D_k}\bar{p}_{dk}^-. \tag{49}$$

Here, $(o_k^*, d_k^*, e_k^*) \in \arg\min_{o\in O_k, d\in D_k, e\in E_{odk}} \bar{\bar{t}}_{odek}$ (a tie is broken arbitrarily), where $\bar{p}_{ok}^+ = \max(0,\bar{p}_{ok})$ and $\bar{p}_{dk}^+ = \max(0,\bar{p}_{dk})$ for all $k\in K$, $o\in O_k$, $\bar{p}_{ok}^- = \min(0,\bar{p}_{ok})$ and $\bar{p}_{dk}^- = \min(0,\bar{p}_{dk})$ for all $k\in K$, $d\in D_k$, and $\bar{\bar{t}}_{odek} = \bar{t}_{odek} + \bar{p}_{ok}^+ + \bar{p}_{dk}^+$ for all $o\in O_k$, $d\in D_k$, $e\in E_{odk}$. The optimal objective value of the Lagrangian subproblem $L(\psi,\xi)$, denoted by $\hat{L}(\psi,\xi)$, is then given by

$$\hat{L}(\psi,\xi) = \hat{L}_1(\psi) + \sum_{k\in K}\hat{L}_2^k(\psi,\xi). \tag{50}$$

Proof. Since $L_1(\psi)$ is an unconstrained linear optimization problem with binary decision variables, it is straightforward to obtain its optimal solution in Eq. (44) and objective value in Eq. (45).

Now, consider $L_2^k(\psi, \xi)$. For any $\bar{p}_{ok} \leq 0$ ($\bar{p}_{dk} \leq 0$), the optimal solution for the corresponding y_{ok} (u_{dk}) should be 1, since $y_{ok} = 1$ ($u_{dk} = 1$) will make Eq. (40) (Eq. (41)) a dummy constraint. This corresponds to the term $\sum_{o \in O_k} \bar{p}_{ok}^-$ ($\sum_{d \in D_k} \bar{p}_{dk}^-$) in the optimal objective (49). For any $\bar{p}_{ok} > 0$ ($\bar{p}_{dk} > 0$), it would be preferable to choose the corresponding $y_{ok} = 0$ ($u_{dk} = 0$) so as to minimize the objective. However, it might be impossible to set all these y_{ok} (u_{dk}) to 0 due to constraints (39)–(41). Thus, the optimal solution is to choose one of the x_{odek} to be 1 and all others to be 0, such that $\bar{t}_{odek} + \bar{p}_{ok}^+ + \bar{p}_{dk}^+$ is minimized, leading to conclusions in (46)–(49).

Finally, since $L(\psi, \xi)$ is decomposed to $L_1(\psi, \xi)$ and $L_2(\psi, \xi)$, which is further decomposed to $|K|$ subproblems $L_2^k(\psi, \xi)$, it is not difficult to see that Eq. (50) holds. □

4.1.2. Subgradient Procedure. The solution of the Lagrangian subproblems, for any choice of the Lagrange multipliers (ψ, ξ), provides a lower bound to the FHLP. To obtain the best lower bound, we must solve the Lagrangian dual:

$$(D_{FHLP}): \quad \max_{(\psi,\xi) \geq 0} L(\psi, \xi).$$

We apply subgradient optimization to solve the Lagrangian dual problem D_{FHLP}. In the subgradient optimization procedure, we exploit the primal solutions obtained from the Lagrangian subproblem to construct feasible solutions and upper bounds to the FHLP. Before the description of the Lagrangian relaxation heuristic, we introduce the following notation:

$\epsilon, \bar{\epsilon}$	Current iteration count and maximum number of iterations allowed.
LB, UB	Incumbent lower bound and upper bound for the FHLP.
δ	Tolerance between LB and UB.
$(\psi^\epsilon, \xi^\epsilon)$	Dual multipliers (ψ, ξ) at iteration ϵ.
λ^ϵ	Step size of movement for the subgradient procedure at iteration ϵ.
ω	Predefined parameter.
$\ell, \bar{\ell}$	Current and maximum number of consecutive iterations without a lower bound improvement.
ϕ	Shrinkage parameter for updating ω.

With the above notation, the proposed subgradient-based Lagrangian relaxation heuristic is provided in Algorithm 2.

The algorithm for constructing a feasible solution $(\ddot{z}^\epsilon, \ddot{x}^\epsilon, \ddot{y}^\epsilon, \ddot{u}^\epsilon)$ of FHLP from the solution of the Lagrangian subproblem in iteration ϵ is given in Algorithm 3. First, the algorithm sets the values of $(\ddot{z}_h^\epsilon, \ddot{y}_{ok}^\epsilon, \ddot{u}_{dk}^\epsilon)$ to be one if there is any flow going through hub h, origin node o, and destination node d for commodity k based on solution values $\hat{x}_{odek}^\epsilon$ of the Lagrangian subproblem; otherwise the value is set to zero. Next, the algorithm constructs solution values $(\ddot{x}_{odek}^\epsilon, \ddot{y}_{ok}^\epsilon, \ddot{u}_{dk}^\epsilon)$ for each commodity based on capacities, available hubs determined by $\ddot{z}_h^\epsilon$, and the ranking of the total assignment and transportation costs for transporting unit flow on each available route. For each commodity $k \in K$, the algorithm checks if the current origin and destination determined by $(\ddot{y}_{ok}^\epsilon, \ddot{u}_{dk}^\epsilon)$ have enough capacities to satisfy its demand, and

Algorithm 2: Lagrangian Relaxation Heuristic

$LB \leftarrow -\infty$, $UB \leftarrow +\infty$, $\epsilon \leftarrow 1$, $\psi^{\epsilon} \leftarrow 0$, $\xi^{\epsilon} \leftarrow 0$, $\omega \leftarrow 2$;
repeat
  Solve $L(\psi^{\epsilon}, \xi^{\epsilon})$ using Proposition 4.2 to obtain the optimal objective value $\hat{L}(\psi^{\epsilon}, \xi^{\epsilon})$ and the corresponding optimal solution $(\hat{z}^{\epsilon}, \hat{x}^{\epsilon}, \hat{y}^{\epsilon}, \hat{u}^{\epsilon})$;
  if $LB < \hat{L}(\psi^{\epsilon}, \xi^{\epsilon})$ **then**
    $LB \leftarrow \hat{L}(\psi^{\epsilon}, \xi^{\epsilon})$;
    $\ell \leftarrow 0$;
  else
    $\ell \leftarrow \ell + 1$;
    if $\ell = \bar{\ell}$ **then**
      $\omega \leftarrow \frac{\omega}{\phi}$;
      $\ell \leftarrow 0$;
  end
  Construct a feasible solution of FHLP $(\ddot{z}^{\epsilon}, \ddot{x}^{\epsilon}, \ddot{y}^{\epsilon}, \ddot{u}^{\epsilon})$ from $(\hat{z}^{\epsilon}, \hat{x}^{\epsilon}, \hat{y}^{\epsilon}, \hat{u}^{\epsilon})$, and compute its corresponding objective value $z(\ddot{z}^{\epsilon}, \ddot{x}^{\epsilon}, \ddot{y}^{\epsilon}, \ddot{u}^{\epsilon})$, using Algorithm 3 ;
  if $UB > z(\ddot{z}^{\epsilon}, \ddot{x}^{\epsilon}, \ddot{y}^{\epsilon}, \ddot{u}^{\epsilon})$ **then**
    $UB \leftarrow z(\ddot{z}^{\epsilon}, \ddot{x}^{\epsilon}, \ddot{y}^{\epsilon}, \ddot{u}^{\epsilon})$;
  Update $(\psi^{\epsilon+1}, \xi^{\epsilon+1})$ using the subgradient procedure in Algorithm 4 ;
  $\epsilon \leftarrow \epsilon + 1$;
until $UB - LB \leq \delta$ *or* $\epsilon = \bar{\epsilon}$;

the solution value $\ddot{x}^{\epsilon}_{odek}$ is set to $\hat{x}^{\epsilon}_{odek}$ if the capacities are enough. Otherwise, the solution value $\ddot{x}^{\epsilon}_{odek}$ is set to a fraction of $\hat{x}^{\epsilon}_{odek}$ based on the ratio between the demand and capacities. Then, a new origin node, destination node, and route with the minimum assignment and transportation costs per-unit flow are assigned to transport the remaining flow, with the exact amount determined by the remaining capacities in the assigned origin and destination. Such a procedure continues until the demand for commodity k is fully satisfied.

At iteration ϵ, a subgradient of $L(\psi^{\epsilon}, \xi^{\epsilon})$ is given by

$$\varphi(\psi^{\epsilon}, \xi^{\epsilon}) = \Bigg(\Big(\sum_{o \in O_k} \sum_{d \in D_k} \sum_{e \in E_{odk}: h \in e} \hat{x}^{\epsilon}_{odek} - \hat{z}^{\epsilon}_{h} \Big)_{h,k}, \Big(\sum_{d \in D_k} \sum_{e \in E_{odk}} w_k \cdot \hat{x}^{\epsilon}_{odek} - c_{ok} \cdot \hat{y}_{ok} \Big)_{o,k}, \Big(\sum_{o \in O_k} \sum_{e \in E_{odk}} w_k \cdot \hat{x}^{\epsilon}_{odek} - c_{dk} \cdot \hat{u}_{dk} \Big)_{d,k} \Bigg).$$

The classical subgradient algorithm utilizes the subgradient of the current iteration to calculate the direction of movement, i.e., $d^{\epsilon} = \varphi(\psi^{\epsilon}, \xi^{\epsilon})$. However, the classical subgradient algorithm often suffers from slow convergence when solving the Lagrangian dual problem. Hence, we apply a deflected subgradient method (e.g., Camerini et al. (1975) and Contreras et al. (2011b)) to enhance the convergence of the algorithm. The deflected subgradient method uses the deflection parameter π^{ϵ} and the direction of the previous

Algorithm 3: Smoothing Algorithm for Obtaining a Feasible Solution of FHLP $(\ddot{z}^\epsilon, \ddot{x}^\epsilon, \ddot{y}^\epsilon, \ddot{u}^\epsilon)$

Input: $(\hat{z}^\epsilon, \hat{x}^\epsilon, \hat{y}^\epsilon, \hat{u}^\epsilon)$;
Initialize $(\ddot{z}^\epsilon, \ddot{x}^\epsilon, \ddot{y}^\epsilon, \ddot{u}^\epsilon) = (0, \hat{x}^\epsilon, 0, 0)$, $w'_k = w_k$, $p'_{ok} = p_{ok}$, $p'_{dk} = p_{dk}$, $c'_{ok} = c_{ok}$, and $c'_{dk} = c_{dk}$;
Initialize $\beta_k = 0$, $\beta'_k = 0$, $\beta''_k = 0$, and $\mu = 0$;
Set $\ddot{z}^\epsilon_h = 1$ if $\sum_{o\in O_k}\sum_{d\in D_k}\sum_{e\in E_{odk}: h\in e} \hat{x}^\epsilon_{odek} > 0$, $\forall\, h \in H$;
Set $\ddot{y}^\epsilon_{ok} = 1$ if $\sum_{d\in D_k}\sum_{e\in E_{odk}} \hat{x}^\epsilon_{odek} > 0$, $\forall\, k \in K$, $o \in O_k$;
Set $\ddot{u}^\epsilon_{dk} = 1$ if $\sum_{o\in O_k}\sum_{e\in E_{odk}} \hat{x}^\epsilon_{odek} > 0$, $\forall\, k \in K$, $d \in D_k$;
for $k \in K$ **do**
  Set $\mathbb{E} = \{(o \in O_k, d \in D_k) | \ddot{y}^\epsilon_{ok} = 1 \text{ and } \ddot{u}^\epsilon_{dk} = 1\}$, $\mathbb{F} = \{(o \in O_k, d \in D_k, e \in E_{odk}) | \ddot{x}^\epsilon_{odek} = 1\}$;
  while $\top$ **do**
    for $(o, d) \in \mathbb{E}$ **do**
      Set $\ddot{y}^\epsilon_{ok} = 1$, $\ddot{u}^\epsilon_{dk} = 1$, $p'_{ok} = 0$, and $p'_{dk} = 0$;
      Set $\beta'_k = \frac{w'_k}{c'_{ok}}$, $\beta''_k = \frac{w'_k}{c'_{dk}}$, $\beta_k = \max(\beta'_k, \beta''_k)$;
      if $\beta_k > 1$ **then**
        Set $w'_k = w'_k - \frac{w'_k}{\beta_k}$, $c'_{ok} = c'_{ok} - \frac{c'_{ok}}{\beta_k}$, $c'_{dk} = c'_{dk} - \frac{c'_{dk}}{\beta_k}$;
        Set $\ddot{x}^\epsilon_{odek} = \frac{1}{\beta_k} \cdot \frac{w'_k}{w_k}$, $\forall\, (o, d, e) \in \mathbb{F}$;
        else
          Set $\ddot{x}^\epsilon_{odek} = \frac{w'_k}{w_k}$, $\forall\, (o, d, e) \in \mathbb{F}$;
          break ;
        end
      end
    end
    Set $\mu = \min_{o\in O_k, d\in D_k, e\in E_{odk}} \left\{ \frac{p'_{ok}+p'_{dk}}{w'_k} + \frac{t_{odek}}{w_k} | \ddot{z}^\epsilon_{e_1} = 1, \ddot{z}^\epsilon_{e_2} = 1, c'_{ok} > 0, c'_{dk} > 0 \right\}$;
    if $(o, d, e) \in \left\{ (o \in O_k, d \in D_k, e \in E_{odk}) | \mu = \frac{p'_{ok}+p'_{dk}}{w'_k} + \frac{t_{odek}}{w_k}, \ddot{z}^\epsilon_{e_1} = 1, \ddot{z}^\epsilon_{e_2} = 1, c'_{ok} > 0, c'_{dk} > 0 \right\}$ **then**
      Set $\mathbb{E} = \{(o, d)\}$ and $\mathbb{F} = \{(o, d, e)\}$;
    end
  end
end
return $(\ddot{z}^\epsilon, \ddot{x}^\epsilon, \ddot{y}^\epsilon, \ddot{u}^\epsilon)$

Note: Symbol $\top$ means logical true.

iteration to compute the current direction of movement, i.e., $d^\epsilon = \varphi(\psi^\epsilon, \xi^\epsilon) + \pi^\epsilon \cdot d^{\epsilon-1}$. The effectiveness of the deflected subgradient method depends on the setting of π^ϵ, and the literature has investigated several choices of the deflection parameter (e.g., Brännlund (1995) and Sherali and Ulular (1989)). We use the rule

$$\pi^\epsilon = \begin{cases} ||\varphi(\psi^\epsilon, \xi^\epsilon)||/||d^{\epsilon-1}||, & \text{if } \varphi(\psi^\epsilon, \xi^\epsilon) \cdot d^{\epsilon-1} < 0, \\ 0, & \text{otherwise,} \end{cases}$$

based on the geometrical arguments (e.g., Camerini et al. (1975) and Contreras et al. (2011b)). An implementation of the deflected subgradient algorithm is described in Algorithm 4.

Algorithm 4: Subgradient Procedure at Iteration ϵ

Initialize $d^0 = 0$ and compute the subgradient $\varphi(\psi^\epsilon, \xi^\epsilon)$;

if $\varphi(\psi^\epsilon, \xi^\epsilon) \cdot d^{\epsilon-1} < 0$ **then**
  $\pi^\epsilon = ||\varphi(\psi^\epsilon, \xi^\epsilon)||/||d^{\epsilon-1}||$

else
  $\pi^\epsilon = 0$

end

Obtain the direction of movement $d^\epsilon = \varphi(\psi^\epsilon, \xi^\epsilon) + \pi^\epsilon \cdot d^{\epsilon-1}$;

if $\epsilon = 1$ **then**
  Compute the step length $\lambda^\epsilon \leftarrow \omega \cdot \frac{\hat{L}(\psi^\epsilon, \xi^\epsilon)}{\varphi(\psi^\epsilon, \xi^\epsilon) \cdot d^{\epsilon-1}}$;

else
  Compute the step length $\lambda^\epsilon \leftarrow \omega \cdot \frac{(UB - \hat{L}(\psi^\epsilon, \xi^\epsilon))}{\varphi(\psi^\epsilon, \xi^\epsilon) \cdot d^{\epsilon-1}}$;

end

$(\psi^{\epsilon+1}, \xi^{\epsilon+1}) \leftarrow (\psi^\epsilon, \xi^\epsilon) + \lambda^\epsilon \cdot d^\epsilon$;

4.2. Cutting Planes

In this section, we introduce cutting planes that can effectively improve the computational speed of the algorithm. In addition to the cutting planes presented in this subsection, Pareto-optimal cuts are proposed to further improve the convergence of the Benders algorithm. However, due to their lower effectiveness in our numerical experiments compared to other cuts, we present them in Section 1 of the Online Supplement to this paper.

4.2.1. Generating Initial Cuts Using Uncapacitated FLHP. The FLHP becomes the uncapacitated FHLP (UFHLP) by dropping constraints (12)–(13) and adding constraints (36)–(37), i.e., UFHLP = $\{\min(9) | (z, x, y, u) \in (10), (11), (14) - (16), (36), (37)\}$. In the Benders reformulation of the UFHLP, the variables (z, y, u) are handled in the master problem, which is defined as $\mathcal{MP}_{\mathcal{U}} = \{\min(28)) | (z, y, u) \in (31) - (35), \zeta \geq \sum_{k\in K} \alpha_k - \sum_{k\in K} \sum_{h\in H} z_h \cdot \theta_{hk} - \sum_{k\in K} \left(\sum_{o\in O_k} y_{ok} \cdot \gamma_{ok} - \sum_{d\in D_k} u_{dk} \cdot \tau_{dk} \right), \ \forall \ (\alpha, \theta, \gamma, \tau) \in \mathbb{P}^e\}$. The variables x are handled in the primal subproblem, which is defined as $\mathcal{PS}_{\mathcal{U}} = \{\min(17) | (x) \in (18) - (19), (22), \sum_{d\in D_k} \sum_{e\in E_{odk}} x_{odek} \leq \tilde{y}_{ok}, \forall \ k \in K, \ o \in O_k, \ \sum_{o\in O_k} \sum_{e\in E_{odk}} x_{odek} \leq \tilde{u}_{dk}, \forall \ k \in K, \ d \in D_k\}$. The dual subproblem ($\mathcal{DS}_{\mathcal{U}}$) can be presented as follows:

$$(\mathcal{DS}_{\mathcal{U}}): \quad \max \left\{ \sum_{k\in K} \alpha_k - \sum_{k\in K} \sum_{h\in H} \tilde{z}_h \cdot \theta_{hk} - \sum_{k\in K} \sum_{o\in O_k} \tilde{y}_{ok} \cdot \gamma_{ok} - \sum_{k\in K} \sum_{d\in D_k} \tilde{u}_{dk} \cdot \tau_{dk} \right\} \tag{51}$$

subject to:

$$\alpha_k - \theta_{e_1 k} - \theta_{e_2 k} - \gamma_{ok} - \tau_{dk} \leq t_{odek}, \quad \forall \ k \in K, \ o \in O_k, \ d \in D_k, \ e \in E_{odk}, \ |e| = 2, \tag{52}$$

$$\alpha_k - \theta_{e_1 k} - \gamma_{ok} - \tau_{dk} \leq t_{odek}, \quad \forall \ k \in K, \ o \in O_k, \ d \in D_k, \ e \in E_{odk}, \ |e| = 1, \tag{53}$$

$$\theta_{hk} \geq 0, \quad \forall \ k \in K, \ h \in H, \tag{54}$$

$$\gamma_{ok} \geq 0, \ \tau_{dk} \geq 0, \quad \forall \ k \in K, \ o \in O_k, \ d \in D_k. \tag{55}$$

Since the UFLHP is a relaxation of the FLHP, we can warm start the Benders decomposition algorithm for the FLHP by first solving the UFLHP to generate an initial set of optimality cuts. The merit of first solving the uncapacitated model is that a large number of optimality cuts can be generated from the dual subproblems that are much smaller and easier to solve. These optimality cuts are valid for the $\mathcal{MP}$ because the primal subproblem for the UFHLP is a relaxation for that of the FHLP achieved by removing the capacity constraints. The polyhedron of the dual subproblem associated with the master problem of the UFLHP is thus contained in that of the dual subproblem associated with the $\mathcal{MP}$. Hence, a dual solution obtained by solving the dual subproblem of the UFHLP is assured to be feasible for the $\mathcal{MP}$. This solution may lie in the interior of the polyhedron but it will nonetheless provide a valid optimality cut. The idea of solving a relaxed problem to generate initial cuts for a Benders decomposition algorithm was implemented before, e.g., by Cordeau et al. (2001). Moreover, it is known that both the lower bound of the master problem in the Benders reformulation of the UFHLP and its optimal solution can yield a valid lower bound for the $\mathcal{MP}$ problem.

For any $k \in K$, let $\tilde{O}_k = \{o \in O_k | \tilde{y}_{ok} = 1\}$ and $\tilde{D}_k = \{d \in D_k | \tilde{u}_{dk} = 1\}$. Further, let $H_0 = \{h \in H | \tilde{z}_h = 0\}$ be the set of non-leased hubs and $H_1 = \{h \in H | \tilde{z}_h = 1\}$ be the set of leased hubs. The primal subproblem $\mathcal{PS}_{\mathcal{U}}$ can be rewritten as $|K|$ independent smaller problems $\mathcal{PS}_{\mathcal{U}k}$ for each $k \in K$,

$$(\mathcal{PS}_{\mathcal{U}k}): \quad \min \left\{ \sum_{o \in \tilde{O}_k} \sum_{d \in \tilde{D}_k} \sum_{e \in E_{odk} \cap (H_1 \times H_1)} t_{odek} \cdot x_{odek} \right\} \tag{56}$$

subject to:

$$\sum_{o \in \tilde{O}_k} \sum_{d \in \tilde{D}_k} \sum_{e \in E_{odk} \cap (H_1 \times H_1)} x_{odek} = 1, \tag{57}$$

$$x_{odek} \geq 0, \quad \forall\, o \in \tilde{O}_k, d \in \tilde{D}_k, e \in E_{odk} \cap (H_1 \times H_1). \tag{58}$$

It is not difficult to observe that the optimal solution of $\mathcal{PS}_{\mathcal{U}k}$ is achieved at $x_{o_k^* d_k^* e_k^* k} = 1$ and $x_{odek} = 0$ for all other values of (o, d, e), where $(o_k^*, d_k^*, e_k^*, k) \in \arg\min_{o \in \tilde{O}_k, d \in \tilde{D}_k, e \in E_{odk} \cap (H_1 \times H_2)} t_{odek}$. The corresponding optimal objective value of $\mathcal{PS}_{\mathcal{U}}$ for $(\tilde{z}, \tilde{y}, \tilde{u})$, denoted by $z(\mathcal{PS}_{\mathcal{U}}(\tilde{z}, \tilde{y}, \tilde{u}))$, can be calculated as follows:

$$z(\mathcal{PS}_{\mathcal{U}}(\tilde{z}, \tilde{y}, \tilde{u})) = \sum_{k \in K} t_{o_k^* d_k^* e_k^* k} = \sum_{k \in K} \min_{o \in \tilde{O}_k, d \in \tilde{D}_k, e \in E_{odk} \cap (H_1 \times H_2)} t_{odek}. \tag{59}$$

4.2.2. Valid Cuts. The following valid cuts (60)–(61) can be added to the master problem to generate solutions that may less likely cause the infeasiblity of the subproblems and accelerate convergence to the optimal solution:

$$\sum_{o \in O_k} y_{ok} \geq \frac{w_k}{\max\limits_{o \in O_k} c_{ok}}, \quad \forall\, k \in K, \tag{60}$$

$$\sum_{d \in D_k} u_{dk} \geq \frac{w_k}{\max\limits_{d \in D_k} c_{dk}}, \quad \forall\, k \in K. \tag{61}$$

Constraints (60)–(61) ensure the minimum number of origin/destination nodes required for commodity k based on the ratio between its demand and the maximum supply/procurement capacity of the commodity. When the demand of a commodity is larger than the maximum supply/procurement capacity of the commodity for all of its origin/destination nodes, these additional constraints can ensure that more than one origin/destination nodes are selected for the commodity, which can significantly boost the solution quality of the master problem passed to the dual subproblem. For each $k \in K$, we rank c_{ok} for all $o \in O_k$ in a decreasing order, with the rank of c_{ok} denoted by $\bar{c}_{ok}$. Then, constraints (60)–(61) can be further enhanced to constraints (62)–(63) as follows:

$$\sum_{o \in O_k} y_{ok} \geq \min\{\hbar | \sum_{o \in O_k | \bar{c}_{ok} \leq \hbar} c_{ok} \geq w_k\}, \quad \forall\, k \in K, \tag{62}$$

$$\sum_{d \in D_k} u_{dk} \geq \min\{\hbar | \sum_{d \in D_k | \bar{c}_{dk} \leq \hbar} c_{dk} \geq w_k\}, \quad \forall\, k \in K. \tag{63}$$

4.3. Clustering-empowered Multi-commodity Benders Reformulation

The number of cuts needed to obtain an optimal solution of the Benders reformulation can equal the number of extreme points in $\mathbb{P}$ in the worst case. However, this number can be reduced given that the subproblem can be decomposed to $|K|$ independent subproblems (see, e.g., Birge and Louveaux (1988)). Hence, we can generate cuts from the dual polyhedra of these $|K|$ subproblems. Let ζ_k be the expected total cost of ζ for commodity k. Let $\mathbb{P}^e_k$ and $\mathbb{P}^u_k$ be the sets of extreme points of $\mathbb{P}^e$ and unbounded rays of $\mathbb{P}^u$ of the subproblem associated with commodity k, respectively. We obtain the following Benders reformulation:

$$(\mathcal{MP}): \quad \min \left\{ \sum_{h \in H} f_h \cdot z_h + \sum_{k \in K} \sum_{o \in O_k} p_{ok} \cdot y_{ok} + \sum_{k \in K} \sum_{d \in D_k} p_{dk} \cdot u_{dk} + \sum_{k \in K} \zeta_m \right\} \tag{64}$$

$$\text{subject to: } (31) - (35),$$

$$\zeta_k \geq \alpha_k - \sum_{h \in H} z_h \cdot \theta_{hk} - \sum_{o \in O_k} c_{ok} \cdot y_{ok} \cdot \gamma_{ok} - \sum_{d \in D_k} c_{dk} \cdot u_{dk} \cdot \tau_{dk}, \quad \forall\, k \in K,\ (\alpha, \theta, \gamma, \tau) \in \mathbb{P}^e_k, \tag{65}$$

$$\alpha_k - \sum_{h \in H} z_h \cdot \theta_{hk} - \sum_{o \in O_k} c_{ok} \cdot y_{ok} \cdot \gamma_{ok} - \sum_{d \in D_k} c_{dk} \cdot u_{dk} \cdot \tau_{dk} \leq 0, \quad \forall\, k \in K,\ (\alpha, \theta, \gamma, \tau) \in \mathbb{P}^u_k. \tag{66}$$

When the size of $|K|$ gets large, adding $|K|$ cuts per iteration can increase the computational effort required for solving the relaxed master problems. Instead of generating all $|K|$ cuts in a disaggregated way at each iteration, we can add a set of feasibility and optimality cuts associated with the clusters of commodity families derived by clustering models. Our motivation here is to use the clustering models to group commodities with similar origin and destination nodes into the same cluster such that the performance of the Benders reformulation can be enhanced. Let $\mathcal{R}$ be the total number of clusters, indexed by r, and C_r be the subset of commodities in any cluster $r \in \mathcal{R}$. Let ζ_r be the expected total cost of ζ for commodity cluster r. Let $\mathbb{P}^e_r$ and $\mathbb{P}^u_r$ be the sets of extreme points of $\mathbb{P}^e$ and unbounded rays of $\mathbb{P}^u$ of the subproblem associated with commodity cluster r, respectively. The multi-commodity cluster Benders reformulation can be presented as follows:

$$(\mathcal{MP}): \min \left\{ \sum_{h \in H} f_h \cdot z_h + \sum_{k \in K} \sum_{o \in O_k} p_{ok} \cdot y_{ok} + \sum_{k \in K} \sum_{d \in D_k} p_{dk} \cdot u_{dk} + \sum_{r \in \mathcal{R}} \zeta_r \right\} \tag{67}$$

subject to: (31) – (35),

$$\zeta_r \geq \sum_{k\in C_r} \alpha_k - \sum_{k\in C_r}\sum_{h\in H} z_h \cdot \theta_{hk} - \sum_{k\in C_r}\left(\sum_{o\in O_k} c_{ok}\cdot y_{ok}\cdot \gamma_{ok} - \sum_{d\in D_k} c_{dk}\cdot u_{dk}\cdot \tau_{dk}\right),\ \forall\ r\in\mathcal{R},\ (\alpha,\theta,\gamma,\tau)\in\mathbb{P}^e_r, \tag{68}$$

$$\sum_{k\in C_r} \alpha_k - \sum_{k\in C_r}\sum_{h\in H} z_h \cdot \theta_{hk} - \sum_{k\in C_r}\left(\sum_{o\in O_k} c_{ok}\cdot y_{ok}\cdot \gamma_{ok} - \sum_{d\in D_k} c_{dk}\cdot u_{dk}\cdot \tau_{dk}\right)\leq 0,\ \forall\ r\in\mathcal{R},\ (\alpha,\theta,\gamma,\tau)\in\mathbb{P}^u_r. \tag{69}$$

Table 1 shows an example of input data for clustering models. The input data are obtained from O_k and D_k. In Table 1, the number "1" for each origin node and commodity indicates that the commodity can originate from the origin node, similarly for the numbers associated with destination nodes.

Table 1 Input Data for Clustering Models

	O_k								D_k							
k\(o, d)	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
1	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0
2	0	1	1	0	0	0	0	0	0	0	1	1	0	0	0	0
3	0	0	0	1	1	0	0	0	0	0	1	1	1	0	0	0
4	0	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0
5	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1
6	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1

Different portions of the cost data could be used to build the clustering models based on their patterns. For example, when the average cost of p_{ok} is much higher than the average cost of p_{dk}, we can build the clustering models using the input data associated with O_k. When the assignment costs p_{ok} and p_{dk} are close, we can build the clustering models using the input data associated with both O_k and D_k. Examples of these clustering variants are illustrated in Figure 2.

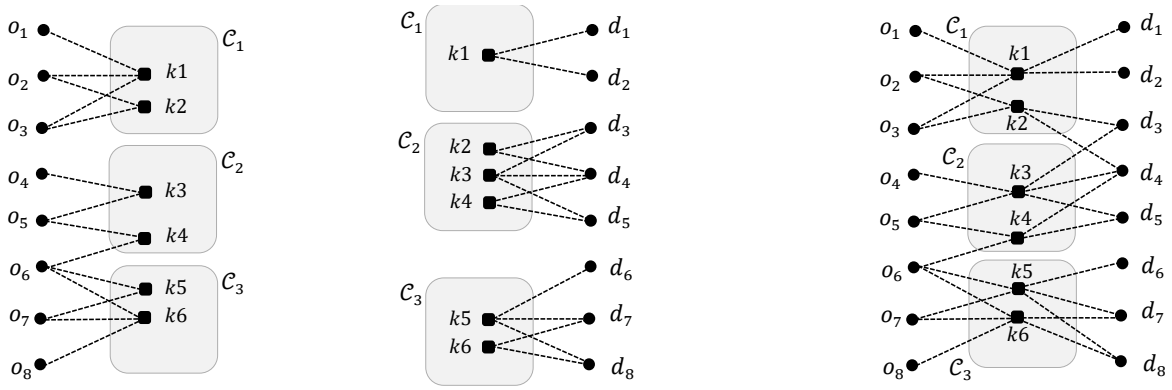

Figure 2 Clustering Variants

Regarding the clustering technique, a customized K-means clustering model is used. Specifically, let the number of clusters be an input variable and use a threshold κ on the average distance within each cluster as the stopping criterion. The clustering procedure begins by classifying the commodities into a given

number of clusters using the K-means method. The average distance within each cluster is then calculated. The clustering procedure stops if the average distance within each cluster is smaller than the threshold κ. Otherwise, it increases the number of clusters and reruns the aforementioned K-means clustering. This process repeats until the average distance within each cluster satisfies the stop criterion.

4.4. Learning-empowered Elimination Tests (LET) and Variable Reduction (LVR)

Along with the variable-reduction technique discussed in Section 3.2, we can use the LET to further reduce the size of the master problem and subproblems by exploiting the information obtained during the inner iterations of the Benders decomposition algorithm, similar to the two elimination tests studied by Contreras et al. (2011a) and Taherkhani et al. (2020).

More specifically, the following two elimination tests can be performed.

1. Determine a set of hubs $Q_0 \subset H$ that must be closed in an optimal solution using $\mathcal{UB}$ and the primal information obtained by solving the LP relaxation of $\mathcal{MP}$. Specifically, let $\mathcal{MP}^{\varepsilon}_{LP}$ denote the LP relaxation of $\mathcal{MP}$ at iteration ε, let $z(\mathcal{MP}^{\varepsilon}_{LP})$ denote its optimal objective value, and let rc_h denote the reduced cost associated with nonbasic variable z_h for all $h \in H$. If $z(\mathcal{MP}^{\varepsilon}_{LP}) + rc_h > \mathcal{UB}$, hub h must be closed in an optimal solution, since $z(\mathcal{MP}^{\varepsilon}_{LP}) + rc_h$ is a lower bound on the objective value if hub h is open.

2. Determine a set of hubs $Q_1 \subset H$ that must be closed in an optimal solution using a modified master problem $\mathcal{MP}^{\varepsilon}(Q_1)$. Specifically, the modified master problem $\mathcal{MP}^{\varepsilon}(Q_1)$ can be constructed by adding a constraint $\sum_{h \in Q_1} z_h \geq 1$ to $\mathcal{MP}^{\varepsilon}$. If $z(\mathcal{MP}^{\varepsilon}(Q_1)) > \mathcal{UB}$, all hubs in Q_1 must be closed in an optimal solution, since $z(\mathcal{MP}^{\varepsilon}(Q_1))$ is a lower bound on the objective function value if a hub is located in Q_1.

The variables for hubs that satisfy the above elimination tests, as well as their associated variables and constraints, can be eliminated permanently from the master problem and subproblems, so as to reduce the solution space and speed up the algorithm's convergence.

The performance of the second elimination test depends on the choice of Q_1. Contreras et al. (2011a) chose Q_1 by using the reduced cost associated with nonbasic variables of $\mathcal{MP}^{\varepsilon}_{LP}$. Taherkhani et al. (2020) proposed different ways to generate Q_1 by sorting the hubs in a nondecreasing order of their ratio of fixed cost to capacity and by sorting hubs in a nonincreasing order of the number of times the hubs have been opened in previous iterations of the Benders algorithm. Additionally, rather than running a single test at each iteration, Taherkhani et al. (2020) partitioned Q_1 into a certain number of smaller subsets and ran a reduction test for each subset. In contrast, the LET proposed in this paper chooses Q_1 using likelihood values derived from machine learning models. Let $\mathcal{L}(z)$ denote the likelihood of variables z taking a value of one in an optimal solution. The LET is only performed on hubs that have likelihood values smaller than a threshold χ_1, and the detailed procedure is presented in Algorithm 5.

Besides $\mathcal{L}(z)$, we also introduce $\mathcal{L}(y)$ and $\mathcal{L}(u)$ denoting the likelihood of variables y and u taking a value of one in an optimal solution, respectively, for the LVR technique. We next proceed to introducing the machine learning models used to compute $\mathcal{L}(z)$, $\mathcal{L}(y)$, and $\mathcal{L}(u)$. Such learning-based models are typically built using easily accessible solution values (e.g., Lagrangian relaxation solution values) and problem features. In this paper, we use logistic regression. To generate sufficient data for training the

Algorithm 5: LET for Obtaining Q_1

Input: $\mathcal{UB}$, $\mathcal{L}(z)$, χ_1, and Q_0 ;
termination $\leftarrow$ false ;
$Q_1 \leftarrow H \setminus Q_0$;
repeat
  $Q_1 \leftarrow Q_1 \setminus \{h : \mathcal{L}(z_h) \geq \chi_1\}$;
  if $Q_1 \neq \emptyset$ **then**
    Solve $\mathcal{MP}^{\varepsilon}(Q_1)$ to obtain $z(\mathcal{MP}^{\varepsilon}(Q_1))$;
  if $z(\mathcal{MP}^{\varepsilon}(Q_1)) > \mathcal{UB}$ *or* $Q_1 = \emptyset$ **then**
    termination $\leftarrow$ true ;
  else
    $Q_1 \leftarrow Q_1 \setminus \{h :$ solution of z_h for $\mathcal{MP}^{\varepsilon}(Q_1) = 1\}$;
  end
until *termination = true*;

model, a large number (B) of small-sized instances were created, where their optimal solution values, denoted by $(\hat{z}, \hat{y}, \hat{u}, \hat{x})$, were easily computed. The following is then used.

1. For each small-sized instance,

(a) obtain subgradient-based Lagrangian relaxation solution values for 50 iterations, and compute the average solution values for each specific variable, denoted by $(z_{Lag}, y_{Lag}, u_{Lag}, x_{Lag})$;

(b) obtain LP-relaxation solution values, denoted by $(z_{LP}, y_{LP}, u_{LP}, x_{LP})$;

(c) obtain the solution values of the Benders reformulation for every iteration up to 10 iterations, denoted by $(z^{\varepsilon}_{BD}, y^{\varepsilon}_{BD}, u^{\varepsilon}_{BD})$, as well as the reduced costs for hub variables associated with the solutions of the relaxed master problems, ranked in a nondecreasing order and denoted by $(z^{\varepsilon}_{rcr}, y^{\varepsilon}_{rcr}, u^{\varepsilon}_{rcr})$, $\forall\ \varepsilon \in \{1, ..., 10\}$;

(d) rank the hub-leasing costs and the assignment costs p_{ok} and p_{dk} in a nondecreasing order, denoted by (z_{cr}, y_{cr}, u_{cr}).

Consequently, we generate $B \cdot |H|$ records of $(\hat{z}, z_{Lag}, z_{LP}, z^{\varepsilon}_{BD}, z^{\varepsilon}_{rcr}, z_{cr})$, $B \cdot \sum_{k \in K} |O_k|$ records of $(\hat{y}, y_{Lag}, y_{LP}, y^{\varepsilon}_{BD}, y^{\varepsilon}_{rcr}, y_{cr})$, and $B \cdot \sum_{k \in K} |D_k|$ records of $(\hat{u}, u_{Lag}, u_{LP}, u^{\varepsilon}_{BD}, u^{\varepsilon}_{rcr}, u_{cr})$ observation data for each iteration of the Benders decomposition algorithm.

2. Train a logistic regression model for variable z using the above observation data for each iteration ε ($\forall$, $\varepsilon \in \{1, ..., 10\}$) of the Benders decomposition algorithm, where the solution values of hub location variables, $(z_{cr}, z^{\varepsilon}_{rcr}, z_{LP}, z_{Lag}, z^{\varepsilon}_{BD})$, are independent variables, and $\hat{z}$ is the dependent variable. The trained logistic regression model is given as follows:

$$\mathcal{L}(z) = \frac{e^{f(z)}}{1 + e^{f(z)}}, \quad f(z) = b_0 + b_1 \cdot z_{Lag} + b_2 \cdot z_{LP} + b_3 \cdot z^{\varepsilon}_{BD} + b_4 \cdot z^{\varepsilon}_{rcr} + b_5 \cdot z_{cr}, \tag{70}$$

where $b_0, \ldots, b_5$ are coefficients of the models. Similar models, $\mathcal{L}(y)$ and $\mathcal{L}(u)$, can be trained for the assignment variables y and u, respectively. We note that the models trained for the 10^{th} iteration of the Benders decomposition are used for all iterations larger than 10.

Although the elimination tests have been shown effective in solving the HLPs with high hub setup costs (Contreras et al. 2011a, Taherkhani et al. 2020), its effectiveness in FHLP may be hindered due to the existence of leasing costs, assignment costs, and hub capacity constraints in the FHLPs. Therefore, we further propose the LVR technique to speed up the solution process of the Benders algorithm, introduced as follows:

1. Use the likelihood values $\mathcal{L}(z)$, $\mathcal{L}(y)$, and $\mathcal{L}(u)$ to guide the elimination of variables from the master problem. Although the removed variables are not guaranteed to have a value of zero in the optimal solution, the Benders algorithm can still generate valid cuts for the master problem. As a result, the master problem becomes much smaller so that it is much quicker to generate optimality cuts for the master problem. In this phase, the lower bound of such a master problem is not guaranteed to be a valid lower bound of the FLHP, but the generated valid cuts can still potentially close the optimality gap.

2. Terminate the learning-empowered variable reduction process. The master problem then starts to deliver the valid lower bounds, and the algorithm can converge to optimality.

More specifically, let χ_2, χ_3, and χ_4 be three threshold numbers, let Q_2 be a subset of hubs in H that are removed from the master problem based on χ_2, and let Q_3 be a subset of connection arcs in $K\times O$ and $K\times D$ that are removed from the master problem based on χ_3 and χ_4, with $Q_2 = \{h|\{\mathcal{L}(z_h) < \chi_2, \forall h \in H\backslash Q_1\}\}$ and $Q_3 = \{(ok, dk)|\{\mathcal{L}(y_{ok}) < \chi_3, \forall k \in K, o \in O_k, \mathcal{L}(u_{dk}) < \chi_4, \forall k \in K, d \in D_k\}\}$. Let $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1, Q_2, Q_3)$ be the modified master problem $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon)$ with hubs in Q_1 and Q_2 and connection arcs in Q_3 removed. Let $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1)$ and $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon, Q_1)$ be their corresponding master and dual subproblems with hubs in Q_1 removed. Let $\mathcal{LB}_\mathcal{F}$ be a lower bound of the $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1, Q_2, Q_3)$ problem. Further, let $\mathcal{T}_1$ be the computational time limit for the first phase of the algorithm. With these definitions, a pseudocode of the Benders algorithm with learning-empowered elimination tests and variable reduction is presented in Algorithm 6.

5. Computational Experiments

We generated test instances of the FHLP with various sizes for computational experiments, with the hub-and-spoke network inherited from the Australia Post (AP) data introduced by Ernst and Krishnamoorthy (1996). The AP data is based on a postal delivery network in Sydney, Australia, with 200 nodes representing postal districts. When generating these test instances, we took the coordinates of $|N|$ nodes of the hub-and-spoke network in the AP data. When the number $|N|$ is larger than 200, we extrapolated the coordinates of the AP data to fit the larger-sized instances. The sizes of these test instances are shown in Tables 2–8, with the details of their parameter settings described as follows. Specifically, parameters w_k, p_{ok}, p_{dk}, f_h, $|O_k|$, and $|D_k|$ were uniformly drawn from $C_w\cdot[25, 80]$, $C_a\cdot[20, 40]$, $C_a\cdot[20, 40]$, $C_f\cdot[200, 1000]$, $\{2, \ldots, C_o\}$, and $\{2, \ldots, C_d\}$, respectively. The parameter of transportation cost ($t_{odh_1h_2k}$) was set to $C_t \cdot 10^{-4} \cdot v_k \cdot w_k \cdot (\Delta_{oh_1} + C_e \cdot \Delta_{h_1h_2} + \Delta_{h_2d})$. Parameter v_k was sampled from a uniform distribution in the range of $[0.1, 2]$. Capacities c_{ok} and c_{dk} were uniformly drawn from $C_c\cdot[0.95, 1.15]\cdot w_k$. Unless stated otherwise, parameters C_w, C_a, C_f, C_o, C_d, C_t, and C_e were set to 1, 1, 1, 4, 4, 1, and 0.3, respectively, and there are three scenarios for C_c, namely, 0.7, 1.0, and 1.1.

Algorithm 6: Benders with the LET and the LVR technique

$\mathcal{UB} \leftarrow +\infty$, $\varepsilon \leftarrow 1$, $\mathbb{P}^e_\varepsilon \leftarrow \emptyset$, $\mathbb{P}^u_\varepsilon \leftarrow \emptyset$, phase $\leftarrow 1$;

termination $\leftarrow$ false ;

repeat

  $Q_1 \leftarrow$ Start the learning-empowered elimination tests from a certain iteration ;

  Update $\mathcal{L}(z)$, $\mathcal{L}(y)$, and $\mathcal{L}(u)$;

  if *phase* $= 1$ **then**

    $Q_2 \leftarrow \{h | \{\mathcal{L}(z_h) < \chi_2, \forall h \in H \backslash Q_1\}\}$;

    $Q_3 \leftarrow \{(ok, dk) | \{\mathcal{L}(y_{ok}) < \chi_3, \forall k \in K, o \in O_k, \mathcal{L}(u_{dk}) < \chi_4, \forall k \in K, d \in D_k\}\}$;

    Solve $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1, Q_2, Q_3)$ to obtain $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ and $z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1, Q_2, Q_3))$;

    $\mathcal{LB}_\mathcal{F} \leftarrow z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1, Q_2, Q_3))$;

    if *A stopping criterion is satisfied (e.g.,* $\mathcal{LB}_\mathcal{F} + 0.0001 \geq \mathcal{UB}$ *or the total computing time* $> \mathcal{T}_1$*)* **then**

      if $Q_2 \subseteq Q_1$ *and* $Q_3 = \emptyset$ **then**

        $\mathcal{LB} \leftarrow \mathcal{LB}_\mathcal{F}$;

        termination $\leftarrow$ true ;

      else

        phase $\leftarrow 2$;

      end

  else

    Solve $\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1)$ to obtain $(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon)$ and $z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1))$;

    $\mathcal{LB} \leftarrow z(\mathcal{MP}(\mathbb{P}^e_\varepsilon, \mathbb{P}^u_\varepsilon, Q_1))$;

  end

  if $\mathcal{UB} = \mathcal{LB}$ **then**

    termination $\leftarrow$ true ;

  else

    Solve $\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon, Q_1)$;

    if *Unbounded* **then**

      Get unbounded ray $(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon) \in \mathbb{P}^u$;

      $\mathbb{P}^u_{\varepsilon+1} \leftarrow \mathbb{P}^u_\varepsilon \cup \{(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon)\}$;

    else

      Get extreme point $(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon) \in \mathbb{P}^e$ and $z(\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon, Q_1))$;

      $\mathcal{UB} \leftarrow \min \{\mathcal{UB},$

        $z(\mathcal{DS}(\tilde{z}_\varepsilon, \tilde{y}_\varepsilon, \tilde{u}_\varepsilon, Q_1)) + \sum_{h \in H} f_h \cdot \tilde{z}_h + \sum_{k \in K} \sum_{o \in O_k} p_{ok} \cdot \tilde{y}_{ok} + \sum_{k \in K} \sum_{d \in D_k} p_{dk} \cdot \tilde{u}_{dk}$ $\}$;

      $\mathbb{P}^e_{\varepsilon+1} \leftarrow \mathbb{P}^e_\varepsilon \cup \{(\alpha_\varepsilon, \theta_\varepsilon, \gamma_\varepsilon, \tau_\varepsilon)\}$;

    end

  end

  $\varepsilon \leftarrow \varepsilon + 1$;

until *termination = true*;

We compared the proposed Algorithm 6 (LEBD) with the branch-and-cut method (CPX) included in the commercial Cplex solver, as well as the elimination tests proposed by Contreras et al. (2011a) and Taherkhani et al. (2020) (CET and TET), respectively. We implemented all the above methods using a high-level algebraic modeling language (GAMS) that uses IBM ILOG Cplex 12.10 as the LP/IP solver. All computations were run on a PC with Intel® Core (TM) i7-8750H CPU @ 2.2 GHz with 32 GB RAM, with a time limit ranging from 1,000 seconds to 1 hour depending on the sizes of test instances. For the LEBD method, parametes κ, χ_1, χ_2, χ_3, and χ_4 were set to 1, 0.1, 0.1, 0, and 0, respectively. The computational time allowed to solve the master problem was set to 100 seconds. The time limit $\mathcal{T}_1$ for the first phase is set to one third of the total time allowed for the LEBD method. The method terminates once the gap between the lower and upper bounds is less than 10^{-4}. For the Lagrangian relaxation heuristic used to generate the initial solution value for the LEBD method, parameters ω, ϕ, and $\bar{\ell}$ were set to 2, 1.001, and 20, respectively. The maximum number of iterations allowed is up to 50, and the gap tolerance between lower and upper bounds was set to 10^{-4}. The same settings used for the CET and the TET were applied in our computational experiments. To compare closely with the CET and the TET, we have set parameters χ_3 and χ_4 to be zero such that no connection arcs are removed in Q_3 for the LEBD method in our computational experiments. To further remove connection arcs in Q_3, we suggest that the removal should be more focused on the connection arcs for commodities with more origin and destination options and lower likelihood values $\mathcal{L}(y)$ and $\mathcal{L}(u)$.

Section 2 of the Online Supplement to this paper presents the training and outputs for the logistic regression models proposed in Section 4.4. In summary, the model outputs show that all independent variables are strongly correlated with the dependent variable. Hence, the logistic regression models used in our numerical experiments include all independent variables, and were fitted using the statsmodels.logit module in Python.

Computational results are presented in Tables 2–8, where columns #Ins, #NO, Gap, TM, ε, LB, and UB represent the total number of instances, the number of instances not solved to optimality, the average optimality gap, the average computational time in seconds, the average number of iterations, the lower bound, and the upper bound, respectively. The optimality gaps reported for all the methods except for HEUR are computed as the difference between the lower and upper bounds divided by the upper bound. Since HEUR does not obtain a lower bound, we use the lower bound obtained by the LEBD method to compute its optimality gaps.

5.1. Impact of the Algorithmic Enhancements

We first analyze how each algorithmic enhancement proposed in Section 4 impacts the algorithm performance. Specifically, Tables 2 and 3 show the average enhancement results, where column BSC shows the computational results obtained by the Benders algorithm without any algorithmic enhancements, and the other columns provide results associated with the proposed enhancement techniques. The abbreviations LGBD, FC, IC, VC, CMC, LET, LVR, and PO represent the integrated Lagrangian relaxation and Benders decomposition method, the feasibility check (Section 3.4), initial cuts (Section 4.2.1), valid cuts (Section

4.2.2), clustering-empowered multi-commodity Benders reformulation (Section 4.3), learning-empowered elimination tests (Section 4.4), learning-empowered variable reduction (Section 4.4), and Pareto-optimality cuts (Section 1 of the Online Supplement), respectively.

Table 2 Results for Combinations of Algorithmic Enhancements

						BSC				ENH 1				ENH 2				ENH 3				ENH 4			
C_c	$\|O\|$	$\|D\|$	$\|K\|$	$\|H\|$	#Ins	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM
0.7	15	15	15	10	10	0.00%	0	161	57	0.00%	0	160	55	0.00%	0	160	54	0.00%	0	160	54	0.00%	0	132	51
	25	25	25	15	10	0.05%	3	346	389	0.05%	3	348	383	0.05%	3	343	383	0.05%	3	349	377	0.05%	3	301	373
	40	40	40	25	10	0.72%	10	440	1,003	0.70%	10	442	1,002	0.70%	10	448	1,003	0.68%	10	457	1,003	0.65%	10	390	1,003
	60	60	50	50	10	10.69%	10	131	1,004	3.46%	10	357	1,003	3.21%	10	378	1,003	3.10%	10	388	1,004	3.12%	10	296	1,003
1	15	15	15	10	10	0.00%	0	31	3	0.00%	0	31	3	0.00%	0	31	3	0.00%	0	31	3	0.00%	0	31	3
	25	25	25	15	10	0.00%	0	86	28	0.00%	0	85	27	0.00%	0	85	27	0.00%	0	85	26	0.00%	0	82	25
	40	40	40	25	10	0.00%	1	170	342	0.01%	1	172	359	0.01%	1	172	349	0.00%	1	172	334	0.00%	1	170	327
	60	60	50	50	10	2.57%	10	132	1,005	0.21%	3	253	843	0.20%	3	253	829	0.18%	2	257	799	0.19%	2	254	786
1.1	15	15	15	10	10	0.00%	0	22	2	0.00%	0	20	2	0.00%	0	20	2	0.00%	0	9	1	0.00%	0	9	1
	25	25	25	15	10	0.00%	0	64	25	0.00%	0	64	24	0.00%	0	64	25	0.00%	0	25	7	0.00%	0	25	7
	40	40	40	25	10	0.00%	1	173	475	0.01%	1	171	489	0.01%	1	171	486	0.00%	0	41	50	0.00%	0	41	50
	60	60	50	50	10	1.62%	10	126	1,004	0.32%	8	217	947	0.32%	8	218	946	0.00%	0	92	184	0.00%	0	92	185
Total					120	1.31%	45	157	445	0.40%	36	193	428	0.37%	36	195	426	0.33%	26	172	320	0.33%	26	152	318

ENH 1: LGBD, ENH 2: LGBD+FC, ENH 3: LGBD+FC+IC, and ENH 4: LGBD+FC+IC+VC

Table 3 Results for Additional Combinations of Algorithmic Enhancements

						ENH 5				ENH 6				ENH 7				ENH 8			
C_c	$\|O\|$	$\|D\|$	$\|K\|$	$\|H\|$	#Ins	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM
0.7	15	15	15	10	10	0.00%	0	11	1	0.00%	0	11	2	0.00%	0	11	2	0.00%	0	13	2
	25	25	25	15	10	0.00%	0	18	6	0.00%	0	18	8	0.00%	0	18	7	0.00%	0	21	6
	40	40	40	25	10	0.00%	0	22	65	0.00%	0	22	97	0.00%	0	22	73	0.00%	0	26	39
	60	60	50	50	10	0.004%	1	29	366	0.02%	1	29	592	0.004%	1	29	340	0.00%	0	34	216
1	15	15	15	10	10	0.00%	0	17	2	0.00%	0	17	2	0.00%	0	17	2	0.00%	0	19	3
	25	25	25	15	10	0.00%	0	27	7	0.00%	0	26	7	0.00%	0	27	5	0.00%	0	29	7
	40	40	40	25	10	0.00%	0	35	39	0.00%	0	37	20	0.00%	0	36	19	0.00%	0	38	22
	60	60	50	50	10	0.00%	0	44	162	0.00%	0	44	180	0.00%	0	44	159	0.00%	0	48	149
1.1	15	15	15	10	10	0.00%	0	6	1	0.00%	0	6	1	0.00%	0	7	1	0.00%	0	8	1
	25	25	25	15	10	0.00%	0	9	3	0.00%	0	9	2	0.00%	0	9	2	0.00%	0	11	2
	40	40	40	25	10	0.00%	0	9	14	0.00%	0	10	5	0.00%	0	10	6	0.00%	0	12	5
	60	60	50	50	10	0.00%	0	12	33	0.00%	0	11	32	0.00%	0	11	28	0.00%	0	13	30
Total					120	0.00%	1	20	58	0.00%	1	20	79	0.00%	1	20	54	0.00%	0	23	40

ENH 5: LGBD+FC+IC+VC+CMC, ENH 6: LGBD+FC+IC+VC+CMC+PO, ENH 7: LGBD+FC+IC+VC+CMC+LET, and ENH 8: LGBD+FC+IC+VC+CMC+LET+LVR

More specifically, Tables 2-3 show the results for various combinations of enhancements. By comparing different combinations of enhancements, the main observations are summarized below.

- Comparing the results of BSC and ENH1, the LGBD method can improve the solution quality and reduce the computing time compared to the standard Benders algorithm, indicating that the initial solution values provided from the Lagrangian relaxation heuristic can speed up the convergence process, especially for the large-size test instances.

- Comparing the results of ENH1 and ENH2, the FC procedure can reduce computational time by avoiding solving some infeasible subproblems unnecessarily.

• Comparing the results of ENH2 and ENH3, the IC improves the solution quality for high-capacity instances as it can reduce the total number of iterations required for convergence, and the Benders reformulation of the uncapacitated FHLP is easier to solve than those of the FHLP. We should mention that IC is not used for the high capacitated instances.

• Comparing the results of ENH3 and ENH4, the VC does not enhance the solution quality significantly. However, it can reduce the total number of iterations by reducing the number of infeasibility cuts.

• Comparing the results of ENH4 and ENH5, the CMC can significantly improve the solution quality by reducing the total number of iterations required for convergence.

• Comparing the results of ENH5 and ENH6, although the PO technique often improves convergence for various problems by incorporating optimal cuts, it does not seem to be effective for the FHLP instances, especially for the ones with a low capacity.

• Comparing the results of ENH5 and ENH7, the LET technique eliminates some hubs from the master problem and subproblems, and thus improves the solution quality and computational time.

• Comparing the results of ENH7 and ENH8, the LVR technique can significantly improve the solution quality by solving smaller-sized master problems. In fact, all test instances are solved to optimality after incorporating the LVR.

From the above results and analysis, the Benders algorithm with the LGBD, FC, IC, VC, CMC, LET, and LVR enhancement techniques (i.e., ENH 8) is the best combination of techniques developed in this paper, and we will use this combination as the recommended implementation of the LEBD method in the remaining experiments, unless stated otherwise. To further verify this observation, we performed the Tukey-Kramer HSD tests, which indicate that the optimality gaps obtained by ENH 8 are better than those obtained by other combinations with strong statistical significance.

5.2. Sensitivity Analysis

We evaluated the sensitivity of the algorithm performance and solution quality for various parameter settings described in Table 4. Let SET1 denote the set that includes all instances in G1–G7. There are 30 instances in G1 and 60 instances in each group of G2–G7, resulting in a total of 390 instances in SET1.

Table 4 Description of the Instance Groups for Sensitivity Analysis

$\|O\|$	$\|D\|$	$\|K\|$	$\|H\|$	Group	Feature	Setting	
						Low	High
				G1	Default instances		
				G2	Assignment costs	$C_a = 0.1$	$C_a = 10$
				G3	Demand	$C_w = 0.1$	$C_w = 10$
60	60	50	50	G4	EOS	$C_e = 0.1$	$C_e = 0.6$
				G5	Hub costs	$C_f = 0.1$	$C_f = 10$
				G6	Origin and destination cardinality	$(C_o, C_d) = (3, 3)$	$(C_o, C_d) = (6, 6)$
				G7	Transportation costs	$C_t = 0.1$	$C_t = 10$

The computational results obtained by the LEBD method are displayed in Table 5. We observe that the parameter setting of the capacity, hub costs (G5), and the cardinalities of origin and destination options for each commodity (G6) have the most impactful influence on the efficiency of the LEBD method. The

instances with high cardinalities, low capacities, or low hub costs are more challenging to solve. Moreover, the instances with high demand deviation, high transportation costs, and low EOS are also more challenging to solve. The parameter setting of the assignment costs (G2) is less impactful than the others.

Table 5 Computational Results of LEBD for Sensitivity Analysis

		$C_c = 0.7$					$C_c = 1.0$					$C_c = 1.1$				
Type	Value	#Ins	Gap	#NO	ε	TM	#Ins	Gap	#NO	ε	TM	#Ins	Gap	#NO	ε	TM
G1		10	0.00%	0	33.50	216.17	10	0.00%	0	47.50	148.56	10	0.00%	0	13.20	29.84
G2	Low	10	0.00%	0	31.50	125.11	10	0.00%	0	51.10	153.91	10	0.00%	0	12.80	22.90
	High	10	0.00%	0	38.70	210.28	10	0.00%	0	41.90	131.30	10	0.00%	0	16.90	43.14
G3	Low	10	0.00%	0	10.60	12.78	10	0.00%	0	33.50	97.07	10	0.00%	0	9.30	11.40
	High	10	0.47%	3	37.00	803.71	10	0.00%	0	56.10	170.17	10	0.00%	0	14.80	27.07
G4	Low	10	0.02%	1	36.30	406.99	10	0.00%	0	48.30	253.43	10	0.00%	0	23.60	89.06
	High	10	0.00%	0	26.60	153.45	10	0.00%	0	46.20	84.62	10	0.00%	0	12.70	18.59
G5	Low	10	1.32%	10	29.80	1,127.76	10	0.00%	0	48.50	169.01	10	0.00%	0	16.00	46.55
	High	10	0.00%	0	11.90	18.65	10	0.00%	0	41.00	163.79	10	0.00%	0	9.00	13.37
G6	Low	10	0.00%	0	39.60	163.45	10	0.00%	0	47.50	98.89	10	0.00%	0	15.10	20.29
	High	10	2.36%	10	35.80	1,118.88	10	0.00%	0	53.20	395.25	10	0.00%	0	15.90	58.25
G7	Low	10	0.00%	0	10.90	13.03	10	0.00%	0	33.70	99.13	10	0.00%	0	9.00	10.96
	High	10	0.42%	3	36.30	753.25	10	0.00%	0	55.10	165.24	10	0.00%	0	14.60	26.20
Total	–	130	0.35%	27	29.12	394.12	130	0.00%	0	46.43	163.87	130	0.00%	0	14.07	32.12

The computational results obtained by the CPX method are displayed in Table 6. We observe that the parameter setting of the capacity and the cardinalities of origin and destination options for each commodity (G6) have the most impactful influence on the efficiency of the CPX method. The instances with high cardinalities or low capacities are more challenging to solve. Further, we observe that the LEBD method is most advantageous to the CPX method for low-capacitated instances with high setup costs, low transportation costs, or low demand. It is because the instances with these settings have a higher ratio between hub setup and other costs, and the learning-empowered elimination tests and variable reduction are very efficient to handle these settings.

Table 6 Computational Results of CPX for Sensitivity Analysis

		$C_c = 0.7$				$C_c = 1.0$				$C_c = 1.1$			
Type	Value	#Ins	Gap	#NO	TM	#Ins	Gap	#NO	TM	#Ins	Gap	#NO	TM
G1		10	2.65%	10	1007.90	10	0.00%	0	209.48	10	0.00%	0	16.04
G2	Low	10	0.01%	1	612.21	10	0.00%	0	122.41	10	0.00%	0	15.61
	High	10	1.49%	10	1007.65	10	0.01%	1	400.60	10	0.00%	0	16.19
G3	Low	10	2.40%	8	988.99	10	0.00%	0	227.12	10	0.00%	0	14.87
	High	10	0.02%	1	641.23	10	0.00%	0	183.33	10	0.00%	0	14.68
G4	Low	10	3.54%	10	1007.65	10	0.00%	0	227.84	10	0.00%	0	18.45
	High	10	0.82%	8	956.17	10	0.00%	0	122.54	10	0.00%	0	15.19
G5	Low	10	1.48%	8	911.57	10	0.00%	0	175.45	10	0.00%	0	15.10
	High	10	0.32%	6	828.64	10	0.00%	0	182.66	10	0.00%	0	14.14
G6	Low	10	0.49%	3	801.00	10	0.00%	0	65.95	10	0.00%	0	13.25
	High	10	9.04%	10	1013.63	10	0.07%	3	546.25	10	0.00%	0	32.52
G7	Low	10	1.79%	9	995.88	10	0.00%	0	307.00	10	0.00%	0	15.76
	High	10	0.02%	1	581.00	10	0.00%	0	192.71	10	0.00%	0	14.92
Total	–	130	1.85%	85	873.35	130	0.01%	4	227.95	130	0.00%	0	16.67

We further analyzed the performance of the LEBD method under nine different settings of χ_2, which adjusts the percentage of hubs removed from the master problem based on the likelihood values $\mathcal{L}(z)$ at the first phase of the LEBD method. The computational results are presented in Figure 3. Let PG denote the percentage difference between the upper bound achieved by the first phase of the LEBD method and the final upper bound $\mathcal{UB}$, let RR denote the hub-removal ratio computed as $\frac{|Q_2|}{|H|}$, and let FE denote the error ratio of the removed hubs in Q_2, with FE computed as the number of hubs that are removed in Q_2 and have a solution value of one in the optimal solution divided by $|H|$. In Figure 3, the metrics RR and FE are the average performance for all instances in SET1 that are solved to optimality, and the metric PG is the average performance for all instances in SET1. Further, the TM index is computed as the average of computational time consumed for solving the 390 instances under each of the nine settings divided by the average of their computational time when χ_2 is set to 0.0001, with the Gap and #NO indices computed similarly.

From Figure 3, we observe that more than 65% of hubs can be removed from the master problem in the first phase of the LEBD method when χ_2 is equal to or larger than 0.01, while the error ratio FE is as small as 0.1% when χ_2 is set to 0.01 and increases to 1.5% when χ_2 is set to 0.5. It implies that the learning-empowered variable reduction technique can remove a large portion of hubs with a low error ratio. As indicated by the PG metric, the upper bound obtained by the first phase of the LEBD method is about 0.04% away from the final upper bound obtained by the LEBD method, and it increases to about 1.75% when χ_2 is set to 0.5. These results indicate that the LEBD method can generate optimality cuts leading to good upper bounds during the first phase.

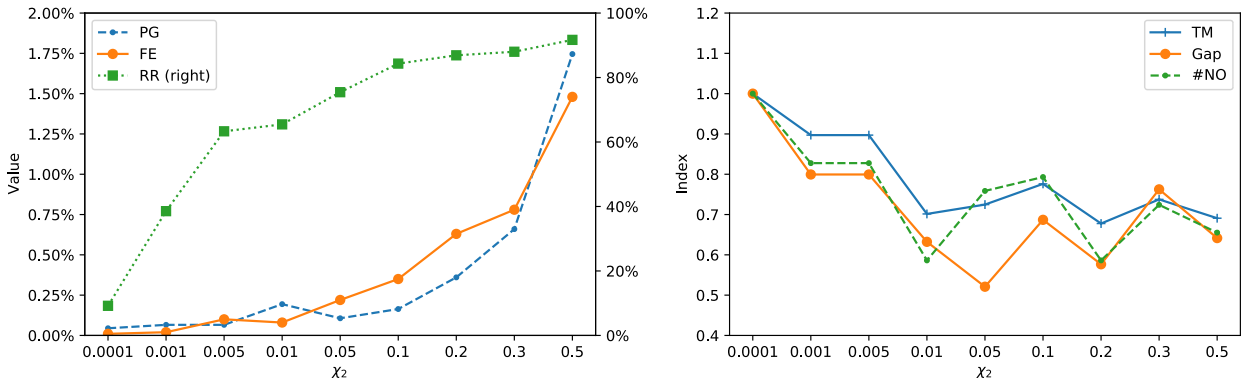

Figure 3 **Performance of the LEBD method under Various Settings of χ_2**

The computational results in Figure 3 also show that the performance indices of TM, Gap, and #NO for the LEBD method improve about 30-40% when parameter χ_2 increases from 0.0001 to 0.01. In this range, the increase of χ_2 can significantly reduce the size of the master problem, while the error ratio FE increases slightly. The performance of the LEBD method starts to fluctuate when the setting of χ_2 ranges from 0.01 to 0.5. As the setting of χ_2 increases within this range, both RR and FE do not change dramatically, and the negative impact from the increase of the error ratio counterbalances the benefit from the size reduction of

the master problem. It explains the fluctuation of the performance of the LEBD method within this range. As such, when deciding the setting of χ_2, we should pay attention to a good balance between the two primary factors, namely, the size of the master problem before and after removing the hubs from Q_2 and the error ratio FE. Because of this, we set parameter χ_2 to be 0.1 as its default value.

5.3. Comparison with Benchmark Algorithms

To compare the proposed algorithm with other benchmark algorithms, we created another set of test instances (SET2) with various sizes described in Table 8. Set SET2 includes 25 instances with low capacities and other parameters set as default. We compare the LEBD method with the CPX, TET, and CET methods by using test instances in SET1 and SET2, with their computational results presented in Tables 7 and 8. The TET and CET methods were implemented based on the LGBD+FC+IC+VC+CMC techniques, such that the differences among TET, CET, and LEBD lie in the learning-empowered elimination test and variable reduction techniques. The computational results presented in Tables 7 and 8 show that the LEBD method outperforms the other three methods, especially for the low-capacitated instances and the ones with large sizes.

Table 7 Solution Comparison for SET1

							CPX			TET				CET				LEBD			
Type	Value	$\|O\|$	$\|D\|$	$\|K\|$	$\|H\|$	#Ins	Gap	#NO	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM	Gap	#NO	ε	TM
G1		60	60	50	50	30	0.88%	10	411.14	0.59%	8	25.37	496.11	0.00%	2	28.33	241.44	0.00%	0	31.40	131.52
C_c	0.70	60	60	50	50	130	1.85%	85	873.35	1.23%	73	18.60	712.00	0.60%	42	22.17	511.49	0.35%	27	29.12	389.50
	1.00	60	60	50	50	130	0.01%	4	227.95	0.22%	7	43.88	305.52	0.10%	2	44.08	158.42	0.00%	0	46.43	163.87
	1.10	60	60	50	50	130	0.00%	0	16.67	0.28%	8	10.82	148.95	0.02%	2	11.06	71.20	0.00%	0	14.07	32.12
G2	Low	60	60	50	50	30	0.00%	1	250.08	1.11%	7	27.60	439.30	0.32%	2	28.93	203.44	0.00%	0	31.80	100.64
	High	60	60	50	50	30	0.50%	11	474.81	0.10%	6	25.53	431.41	0.03%	2	28.43	231.35	0.00%	0	32.50	128.24
G3	Low	60	60	50	50	30	0.80%	8	410.33	0.00%	0	15.90	45.98	0.00%	0	16.10	31.35	0.00%	0	17.80	40.42
	High	60	60	50	50	30	0.01%	1	279.74	0.39%	10	29.10	434.79	0.18%	7	30.07	391.86	0.16%	3	35.97	333.65
G4	Low	60	60	50	50	30	1.18%	10	417.98	1.02%	10	26.03	551.18	0.03%	3	29.60	331.75	0.01%	1	36.07	249.83
	High	60	60	50	50	30	0.27%	8	364.63	0.38%	5	24.57	354.70	0.00%	1	25.67	135.39	0.00%	0	28.50	85.55
G5	Low	60	60	50	50	30	0.49%	8	367.37	0.74%	11	25.77	572.46	0.73%	10	25.77	452.37	0.44%	10	31.43	441.10
	High	60	60	50	50	30	0.11%	6	341.81	0.00%	0	19.43	78.56	0.00%	0	19.43	60.82	0.00%	0	20.63	65.27
G6	Low	60	60	50	50	30	0.16%	3	293.40	0.23%	3	27.30	315.08	0.00%	0	29.33	123.95	0.00%	0	34.07	94.21
	High	60	60	50	50	30	3.04%	13	530.80	2.64%	19	25.27	855.64	1.66%	12	27.07	599.35	0.79%	10	34.97	510.79
G7	Low	60	60	50	50	30	0.60%	9	439.55	0.00%	0	16.17	48.09	0.00%	0	16.20	31.66	0.00%	0	17.87	41.04
	High	60	60	50	50	30	0.01%	1	262.88	0.29%	9	29.60	431.39	0.21%	7	30.10	376.75	0.14%	3	35.33	314.90
Total	–	60	60	50	50	390	0.62%	89	372.66	0.58%	88	24.43	388.82	0.24%	46	25.77	247.04	0.12%	27	29.87	195.17

The LEBD method performs better than the benchmark algorithms for several reasons. First, the LVR technique helps generate many optimality cuts quickly by reducing the size of the master problem in the first phase. Second, unlike the TET and CET techniques that require solving integer programs for the elimination tests, the LEBD method determines Q_2 by using the likelihood values $\mathcal{L}(z)$ that can be derived without a heavy computational burden. Third, the distribution of the likelihood values and its strong correlation with the optimal solution values make it viable to eliminate a significant portion of hubs from H with a relatively low error ratio. To better understand the distribution of the likelihood values $\mathcal{L}(z)$, as well as the correlation between $\mathcal{L}(z)$ and optimal solution values $\hat{z}$, we plot $\mathcal{L}(z)$ and $\hat{z}$ in Figure 4 for all

Table 8 Solution Comparison for SET2

Problem Size					CPX				TET					CET					LEBD				
$\|N\|$	$\|O\|$	$\|D\|$	$\|K\|$	$\|H\|$	LB	UB	Gap	TM	LB	UB	Gap	ε	TM	LB	UB	Gap	ε	TM	LB	UB	Gap	ε	TM
100	40	40	40	20	9,091	9,091	0.00%	30	9,091	9,091	0.00%	14	22	9,091	9,091	0.00%	14	13	9,091	9,091	0.00%	17	9
125	50	50	50	25	12,070	12,070	0.00%	124	12,070	12,070	0.00%	18	98	12,070	12,070	0.00%	18	34	12,070	12,070	0.00%	23	31
150	60	60	60	30	13,146	13,146	0.00%	284	13,146	13,146	0.00%	20	623	13,146	13,146	0.00%	22	181	13,146	13,146	0.00%	24	64
175	70	70	70	35	14,759	14,759	0.00%	560	14,759	14,759	0.00%	27	2,466	14,759	14,759	0.00%	27	538	14,759	14,759	0.00%	29	143
200	80	80	80	40	17,787	17,787	0.00%	2,164	17,654	17,960	1.71%	17	3,600	17,685	17,839	0.87%	18	3,600	17,787	17,787	0.00%	35	959
225	90	90	90	45	20,360	20,381	0.11%	3,600	20,243	20,579	1.63%	17	3,600	20,277	20,579	1.46%	19	3,600	20,381	20,381	0.00%	29	1,094
250	100	100	100	50	21,135	21,906	3.52%	3,600	21,520	22,300	3.50%	16	3,600	21,520	22,300	3.50%	16	3,600	21,803	21,906	0.47%	52	3,600
275	110	110	110	55	23,525	24,465	3.84%	3,600	23,874	24,752	3.55%	16	3,600	23,874	24,752	3.55%	16	3,600	23,997	24,428	1.77%	56	3,600
300	120	120	120	60	24,700	26,507	6.82%	3,600	25,008	27,073	7.63%	10	3,600	25,008	27,073	7.63%	10	3,600	25,695	26,409	2.70%	51	3,600
325	130	130	130	65	26,553	28,614	7.20%	3,600	27,036	29,066	6.99%	11	3,600	27,036	29,066	6.99%	11	3,600	27,706	28,645	3.28%	43	3,600
350	140	140	140	70	28,912	31,188	7.30%	3,600	29,468	30,930	4.73%	12	3,600	29,468	30,930	4.73%	12	3,600	29,704	30,446	2.44%	44	3,600
375	150	150	150	75	30,291	32,691	7.34%	3,600	30,606	33,724	9.25%	10	3,600	30,606	33,724	9.25%	10	3,600	31,277	32,288	3.13%	46	3,600
400	160	160	160	80	31,788	35,080	9.38%	3,600	32,564	35,068	7.14%	10	3,600	32,564	35,068	7.14%	10	3,600	33,437	34,560	3.25%	48	3,600
425	170	170	170	85	33,224	43,196	23.08%	3,600	34,835	38,423	9.34%	11	3,600	34,835	38,423	9.34%	11	3,600	35,541	37,224	4.52%	42	3,600
450	180	180	180	90	34,717	43,521	20.23%	3,600	36,789	40,541	9.26%	10	3,600	36,789	40,541	9.26%	10	3,600	37,718	39,395	4.26%	43	3,600
475	190	190	190	95	35,457	45,504	22.08%	3,600	36,958	40,171	8.00%	10	3,600	36,958	40,171	8.00%	10	3,600	37,988	39,907	4.81%	43	3,600
500	200	200	200	100	37,345	48,019	22.23%	3,600	39,179	43,795	10.54%	8	3,600	39,179	43,795	10.54%	8	3,600	41,149	42,488	3.15%	42	3,600
525	210	210	210	105	40,410	53,190	24.03%	3,600	42,736	47,120	9.30%	9	3,600	42,736	47,120	9.30%	9	3,600	43,905	46,386	5.35%	39	3,600
550	220	220	220	110	41,659	53,420	22.02%	3,600	42,221	49,387	14.51%	7	3,600	42,221	49,387	14.51%	7	3,600	45,359	47,711	4.93%	39	3,600
575	230	230	230	115	42,268	56,719	25.48%	3,600	44,594	50,756	12.14%	9	3,600	44,594	50,756	12.14%	9	3,600	46,445	49,214	5.63%	36	3,600
600	240	240	240	120	44,772	60,197	25.62%	3,600	45,799	55,171	16.99%	7	3,600	45,799	55,171	16.99%	7	3,600	49,505	52,749	6.15%	34	3,600
625	250	250	250	125	46,715	59,920	22.04%	3,600	48,685	55,894	12.90%	8	3,600	48,685	55,894	12.90%	8	3,600	51,653	54,686	5.55%	33	3,600
650	260	260	260	130	48,358	64,708	25.27%	3,600	49,836	59,599	16.38%	7	3,600	49,836	59,599	16.38%	7	3,600	53,401	57,001	6.31%	31	3,600
675	270	270	270	135	49,182	66,814	26.39%	3,600	50,590	61,023	17.10%	7	3,600	50,590	61,023	17.10%	7	3,600	54,472	58,232	6.46%	30	3,600
700	280	280	280	140	51,011	66,055	22.78%	3,600	54,153	62,435	13.26%	8	3,600	54,153	62,435	13.27%	8	3,600	57,192	60,623	5.66%	28	3,600
Average					31,169	37,958	13.07%	3,006	32,137	35,793	7.83%	12	3,152	32,139	35,789	7.79%	12	3,055	33,407	34,861	3.19%	37	2,828

instances in SET1 that are solved to optimality, in a nondecreasing order of $\mathcal{L}(z)$. The results show that the likelihood values are strongly correlated with the optimal solution values, indicating the effectiveness of using the likelihood values to eliminate hubs in the first phase of the LEBD method.

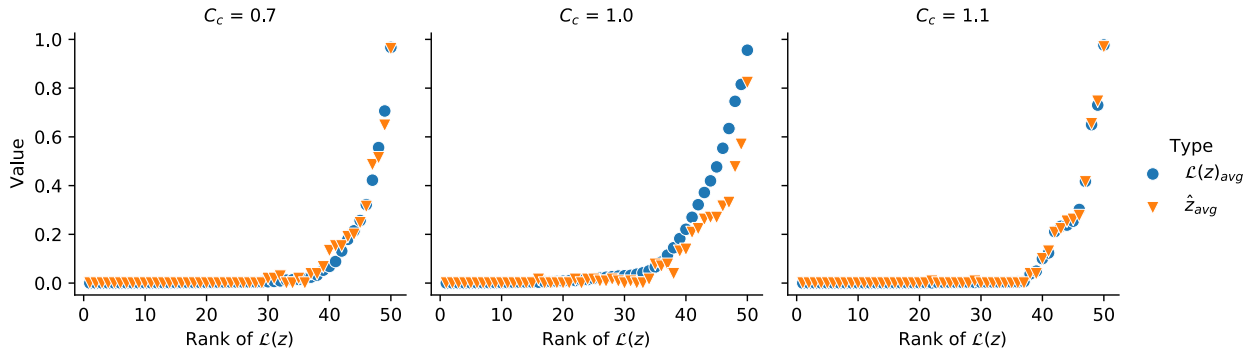

Figure 4 Likelihood Value $\mathcal{L}(z)$ VS Optimal Solution Value $\hat{z}$

The efficiencies of the TET and CET methods are contingent on problem characteristics. Their elimination tests may be efficient in specific situations where hub setup costs are high relative to the other system costs, and an effective Benders process exists to close up the optimality gaps quickly for the elimination tests to take effect. The uncapacitated HLPs often possess these characteristics, so the TET and CET methods can efficiently reduce their hub size and quickly solve large-scale instances. However, the FHLP may not have these characteristics. First, the e-commerce businesses tend to lease hubs, so the hub setup costs are generally low. Second, the assignment costs may take a significant portion of the overall system cost, diluting the chance of hubs being eliminated from the elimination tests. Third, the capacity

constraints make the FHLP more difficult to solve, and it takes more iterations for the Benders algorithm to close gaps for the elimination tests to take effect, which is especially true for large-size instances. Because the elimination tests require solving integer programs, if the success rates of these tests are low or the integer programs are difficult to solve, the elimination tests could lead to a low efficiency for the Benders algorithm.

6. Conclusion and Future Research Directions

We propose a flow hub location (FHL) problem motivated by recent trends in the network design for e-commerce businesses. Specifically, the FHLPs extend the classical HLPs, and add the decisions for choosing origins and destinations simultaneously as the hub leasing and commodity routing decisions are made. Due to the flexibility of e-commerce companies in choosing their hub locations via the leasing option, the size of the FHLPs under study can become quite large. As such, we develop an efficient optimization algorithm that combines machine learning, Lagrangian relaxation, and Benders decomposition to solve this problem. This algorithm applies the learning-empowered Benders decomposition method to a strong path-based formulation. The proposed learning-empowered Benders decomposition method improves from its standard counterpart by exploring several problem-specific properties, including a clustering-empowered multi-commodity Benders reformulation, learning-empowered elimination tests, variable reduction, and the inclusion of the Lagrangian relaxation heuristic for generating good initial solutions. The algorithm's efficiency has been demonstrated via extensive computational tests, where the proposed optimization algorithm outperforms the three benchmark algorithms.

One area of future research is to extend the FHLP by considering various features commonly applied to the hub location problems. Particularly, we are interested in the FHLP combined with demand uncertainty, reliability, or profit maximization with multiple assignments. We are also interested in extending the origin/destination node assignment decisions to the p-hub median problem, the intermodal hub location problem, the cycle hub location problem, and the hub location problems with multiple periods. Further, the FHLPs considered in this paper take deterministic inputs, while uncertainties may be observed in the network design for e-commerce businesses. For example, by choosing different suppliers as the origins and different customer zones as the destinations, the network could be subject to different levels of demand and supply uncertainties. It would be interesting to consider such uncertainties in the network design process.

Another area of future research is to explore heuristics and exact solution approaches to solve variants of the FHLPs. Heuristics of interest include, but are not limited to, tabu search, greedy randomized adaptive search, neighborhood search, and evolutionary-inspired algorithms. Exact methods of interest include branch-and-cut and branch-and-price. We are particularly interested in implementing a machine learning-based optimization framework (e.g., Zhang et al. (2020) and Wu et al. (2022a)) for solving the capacitated FHLPs. The machine learning-based optimization framework aims to build analytical models to gain knowledge on the FHLPs and use this knowledge to fix decision variables (e.g., the hub-leasing decisions) in order to create smaller-sized subproblems, which are then solved to improve solution qualities for the FHLPs progressively.

References

Alumur, S. A., J. F. Campbell, I. Contreras, B. Y. Kara, V. Marianov, M. E. O'Kelly. 2021. Perspectives on modeling hub location problems. *European Journal of Operational Research* **291** 1–17.

Alumur, S. A., S. Nickel, F. Saldanha da Gama. 2012. Hub location under uncertainty. *Transportation Research Part B: Methodological* **46** 529–543.

Armaghan, A., I. Contreras, E. Fernández. 2018. Exact solution of hub network design problems with profits. *European Journal of Operational Research* **266** 57–71.

Birge, J. R., F. V. Louveaux. 1988. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research* **34** 384–392.

Blanco, V., E. Fernández, Y. Hinojosa. 2023. Hub location with protection under interhub link failures. *INFORMS Journal on Computing* **In print**. doi:10.1287/ijoc.2023.1296.

Brännlund, U. 1995. A generalized subgradient method with relaxation step. *Mathematical Programming* **71** 207–219.

Bütün, Cihan, Sanja Petrovic, Luc Muyldermans. 2021. The capacitated directed cycle hub location and routing problem under congestion. *European Journal of Operational Research* **292** 714–734. doi:10.1016/j.ejor.2020.11.021.

Camerini, P. M., L. Fratta, F. Maffioli. 1975. On improving relaxation methods by modified gradient techniques. *Nondifferentiable optimization* 26–34.

Campbell, J. F. 1994. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research* **72** 387–405.

Campbell, J. F. 1996. Hub location and the p-hub median problem. *Operations Research* **44** 923–935.

Campbell, J. F., A. T. Ernst, M. Krishnamoorthy. 2005. Hub arc location problems: part I—introduction and results. *Management Science* **51** 1540–1555.

Campbell, J. F., M. E. O'Kelly. 2012. Twenty-five years of hub location research. *Transportation Science* **46** 153–169.

Contreras, I., J.-F. Cordeau, G. Laporte. 2011a. Benders decomposition for large-scale uncapacitated hub location. *Operations Research* **59** 1477–1490.

Contreras, I., J.-F. Cordeau, G. Laporte. 2011b. The dynamic uncapacitated hub location problem. *Transportation Science* **45** 18–32.

Contreras, I., J.-F. Cordeau, G. Laporte. 2012. Exact solution of large-scale hub location problems with multiple capacity levels. *Transportation Science* **46** 439–459.

Contreras, I., J. A. Díaz, E. Fernández. 2011c. Branch and price for large-scale capacitated hub location problems with single assignment. *INFORMS Journal on Computing* **23** 41–55.

Contreras, I., E. Fernández. 2014. Hub location as the minimization of a supermodular set function. *Operations Research* **62** 557–570.

Contreras, I., M. Tanash, N. Vidyarthi. 2017. Exact and heuristic approaches for the cycle hub location problem. *Annals of Operations Research* **258** 655–677.

Cordeau, J.-F., F. Soumis, J. Desrosiers. 2001. Simultaneous assignment of locomotives and cars to passenger trains. *Operations research* **49** 531–548.

Correia, Isabel, Stefan Nickel, Francisco Saldanha-da Gama. 2018. A stochastic multi-period capacitated multiple allocation hub location problem: Formulation and inequalities. *Omega* **74** 122–134. doi:10.1016/j.omega.2017.01.011.

de Camargo, R. S., Jr. G. de Miranda, H. P. L. Luna. 2009. Benders decomposition for hub location problems with economies of scale. *Transportation Science* **43** 86–97.

de Camargo, R. S., G. Miranda, H. P. Luna. 2008. Benders decomposition for the uncapacitated multiple allocation hub location problem. *Computers & Operations Research* **35** 1047–1064.

de Sá, E. M., R. S. de Camargo, G. Miranda. 2013. An improved Benders decomposition algorithm for the tree of hubs location problem. *European Journal of Operational Research* **226** 185–202.

Elhedhli, S., H. Wu. 2010. A Lagrangean heuristic for hub-and-spoke system design with capacity selection and congestion. *INFORMS Journal on Computing* **22** 282–296.

Ernst, A. T., H. Hamacher, H. Jiang, M. Krishnamoorthy, G. Woeginger. 2009. Uncapacitated single and multiple allocation p-hub center problems. *Computers & Operations Research* **36** 2230–2241.

Ernst, A. T., M. Krishnamoorthy. 1996. Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science* **4** 139–154.

Ernst, A. T., M. Krishnamoorthy. 1998. An exact solution approach based on shortest-paths for p-hub median problems. *INFORMS Journal on Computing* **10** 149–162.

Farahani, R. Z., M. Hekmatfar, A. B. Arabani, E. Nikbakhsh. 2013. Hub location problems: A review of models, classification, solution techniques, and applications. *Computers & Industrial Engineering* **64** 1096–1109.

Gelareh, S., R. N. Monemi, S. Nickel. 2015. Multi-period hub location problems in transportation. *Transportation Research Part E: Logistics and Transportation Review* **75** 67–94.

Gelareh, S., S. Nickel, D. Pisinger. 2010. Liner shipping hub network design in a competitive environment. *Transportation Research Part E: Logistics and Transportation Review* **46** 991–1004.

Ghaffarinasab, Nader, Alireza Motallebzadeh, Younis Jabarzadeh, Bahar Y. Kara. 2018. Efficient simulated annealing based solution approaches to the competitive single and multiple allocation hub location problems. *Computers & Operations Research* **90** 173–192. doi:10.1016/j.cor.2017.09.022.

Guan, Jian, Geng Lin, Hui-Bin Feng. 2018. A learning-based probabilistic tabu search for the uncapacitated single allocation hub location problem. *Computers & Operations Research* **98** 1–12. doi:10.1016/j.cor.2018.04.020.

Hamacher, H. W., M. Labbé, M. Nickel, T. Sonneborn. 2004. Adapting polyhedral properties from facility to hub location problems. *Discrete Applied Mathematics* **145** 104–116.

He, Y., T. Wu, C. Zhang, Z. Liang. 2015. An improved MIP heuristic for the intermodal hub location problem. *Omega* **57** 203–211.

Kargar, K., A. I. Mahmutoğulları. 2022. Risk-averse hub location: Formulation and solution approach. *Computers & Operations Research* **143** 105760.

Klincewicz, J. G. 1996. A dual algorithm for the uncapacitated hub location problem. *Location Science* **4** 173–184.

Klincewicz, J. G. 2002. Enumeration and search procedures for a hub location problem with economies of scale. *Annals of Operations Research* **110** 107–122.

Labbé, M., H. Yaman, E. Gourdin. 2005. A branch and cut algorithm for hub location problems with single assignment. *Mathematical Programming* **102** 371–405.

Larsen, E., E. Frejinger, B. Gendron, A. Lodi. 2023. Fast continuous and integer l-shaped heuristics through supervised learning. *INFORMS Journal on Computing* **In print**. doi:10.1287/ijoc.2022.0175.

Liu, Junming, Weiwei Chen, Jingyuan Yang, Hui Xiong, Can Chen. 2022. Iterative prediction-and-optimization for e-logistics distribution network design. *INFORMS Journal on Computing* **34** 769–789. doi:10.1287/ijoc.2021.1107.

Maiyar, L. M., J. J. Thakkar. 2019. Modelling and analysis of intermodal food grain transportation under hub disruption towards sustainability. *International Journal of Production Economics* **217** 281–297.

Mayer, G., B. Wagner. 2002. Hublocator: an exact solution method for the multiple allocation hub location problem. *Computer & Operations Research* **29** 715–739.

Meng, Q., X. Wang. 2011. Intermodal hub-and-spoke network design: Incorporating multiple stakeholders and multi-type containers. *Transportation Research Part B: Methodological* **45** 724–742.

Mohammadi, M., P. Jula, R. Tovakkoli-Moghaddam. 2019. Reliable single-allocation hub location problem with disruptions. *Transportation Research Part E: Logistics and Transportation Review* **123** 90–120.

Monemi, R. N., S. Gelareh, A. Nagih, D. Jones. 2021. Bi-objective load balancing multiple allocation hub location: A compromise programming approach. *Annals of Operations Research* **296** 363–406.

O'Kelly, M. E. 1987. A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research* **32** 393–404.

O'Kelly, M. E., D. L. Bryan. 1998. Hub location with flow economies of scale. *Transportation Research Part B: Methodological* **32** 605–616.

Özğun-Kibiroğlu, Ç., M. N. Serarslan, Y. İ. Topcu. 2019. Particle swarm optimization for uncapacitated multiple allocation hub location problem under congestion. *Expert Systems With Applications* **119** 1–19.

Rostami, B., N. Kämmerling, J. Naoum-Sawaya, C. Buchheim. 2021. Stochastic single-allocation hub location. *European Journal of Operational Research* **289** 1087–1106.

Shen, H., Y. Liang, Z. J. M. Shen. 2021. Reliable hub location model for air transportation networks under random disruptions. *Manufacturing & Service Operations Management* **23** 388–406.

Sherali, H. D., O. Ulular. 1989. A primal-dual conjugate subgradient algorithm for specially structured linear and convex programming problems. *Applied Mathematics and Optimization* **20** 193–221.

Taherkhani, G., S. A. Alumur, M. Hosseini. 2020. Benders decomposition for the profit maximizing capacitated hub location problem with multiple demand classes. *Transportation Science* **54** 1446–1470.

Taherkhani, G., S. A. Alumur, M. Hosseini. 2021. Robust stochastic models for profit-maximizing hub location problems. *Transportation Science* **55** 1322–1350.

Torkestani, S. S., S. M. Seyedhosseini, A. Makui, K. Shahanaghi. 2018. The reliable design of a hierarchical multi-modes transportation hub location problems (HMMTHLP) under dynamic network disruption (DND). *Computers & Industrial Engineering* **122** 39–86.

Waleed, N., A. Diabat. 2020. Benders decomposition for multiple-allocation hub-and-spoke network design with economies of scale and node congestion. *Transportation Research Part B: Methodological* **133** 62–84.

Wang, S., Z. Chen, T. Liu. 2020. Distributionally robust hub location. *Transportation Science* **54** 1189–1210.

Wu, T., L. Huang, Z. Liang, X. Zhang, C. Zhang. 2022a. A supervised learning-driven heuristic for solving the facility location and production planning problem. *European Journal of Operational Research* **301** 785–796.

Wu, T., Z. Shi, C. Zhang. 2021. The hub location problem with market selection. *Computers & Operations Research* **127** 105136.

Wu, Tao, Canrong Zhang, Weiwei Chen, Zhe Liang, Xiaoning Zhang. 2022b. Unsupervised learning-driven matheuristic for production-distribution problems. *Transportation Science* doi:10.1287/trsc.2022.1149.

Wu, Yuehui, Ali Gul Qureshi, Tadashi Yamada. 2022c. Adaptive large neighborhood decomposition search algorithm for multi-allocation hub location routing problem. *European Journal of Operational Research* **302** 1113–1127. doi:10.1016/j.ejor.2022.02.002.

Zhang, C., H. Guan, Y. Yuan, W. Chen, T. Wu. 2020. Machine learning-driven algorithms for the container relocation problem. *Transportation Research Part B: Methodological* **139** 102–131.

Zhao, Y., Z. Chen, Z. Zhang. 2023. Distributionally robust chance-constrained p-hub center problem. *INFORMS Journal on Computing* **In print**. doi:10.1287/ijoc.2022.0113.

Submitted to *INFORMS Journal on Computing*
manuscript (Please, provide the mansucript number!)

Online Supplement to
Learning-empowered Benders Decomposition for Flow Hub Location in E-Commerce

This online supplement provides additional methods and results to the paper titled "Learning-empowered Benders Decomposition for Flow Hub Location in E-Commerce".

1. Pareto-optimal Cuts

In addition to the cutting planes proposed in Section 4.2 of the paper, we may further improve the convergence of the Benders algorithm by constructing stronger, undominated cuts, known as Pareto-optimal cuts (Magnanti and Wong 1981). A cut generated from the dual solution $(\alpha^a, \theta^a, \gamma^a, \tau^a)$ is Pareto-optimal if no other cuts generated from the dual solution $(\alpha^b, \theta^b, \gamma^b, \tau^b)$ dominates it, and we say that the cut $(\alpha^a, \theta^a, \gamma^a, \tau^a)$ dominates the cut $(\alpha^b, \theta^b, \gamma^b, \tau^b)$ if and only if $\sum_{k\in K}\alpha^a_k - \sum_{k\in K}\sum_{h\in H} z_h \cdot \theta^a_{hk} - \sum_{o\in O_k}\sum_{k\in K} c_{ok}\cdot y_{ok}\cdot \gamma^a_{ok} - \sum_{d\in D_k}\sum_{k\in K} c_{dk}\cdot u_{dk}\cdot \tau^a_{dk} \geq \sum_{k\in K}\alpha^b_k - \sum_{k\in K}\sum_{h\in H} z_h\cdot\theta^b_{hk} - \sum_{o\in O_k}\sum_{k\in K} c_{ok}\cdot y_{ok}\cdot\gamma^b_{ok} - \sum_{d\in D_k}\sum_{k\in K} c_{dk}\cdot u_{dk}\cdot\tau^b_{dk}$, for all $z\in G_1$, $y\in G_2$, and $u\in G_3$ with strict inequality for at least one point. Let $\mathbb{Q}$ be the polyhedron defined by constraints (31)–(33), $0\leq y_{ok}\leq 1$, $\forall\ k\in K$, $o\in O_k$, $0\leq u_{dk}\leq 1$, $\forall\ k\in K$, $d\in D_k$, and $0\leq Z_h\leq 1$, $\forall\ h\in H$, and let $ri(\mathbb{Q})$ denote the relative interior of $\mathbb{Q}$. To determine a Pareto-optimal cut at iteration ε, we need to solve the following Pareto-optimal subproblem $(\mathcal{PO}_\varepsilon)$:

$$
\begin{aligned}
(\mathcal{PO}_\varepsilon):\quad & \max\left\{\sum_{k\in K}\alpha_k - \sum_{k\in K}\sum_{h\in H}\tilde{z}^0_{h\varepsilon}\cdot\theta_{hk} - \sum_{o\in O_k}\sum_{k\in K} c_{ok}\cdot\tilde{y}^0_{ok\varepsilon}\cdot\gamma_{ok} - \sum_{d\in D_k}\sum_{k\in K} c_{dk}\cdot\tilde{u}^0_{dk\varepsilon}\cdot\tau_{dk}\right\}\\
& \text{subject to:}\\
& (19)-(22),\\
& \sum_{k\in K}\alpha_k - \sum_{k\in K}\sum_{h\in H}\tilde{z}_{h\varepsilon}\cdot\theta_{hk} - \sum_{o\in O_k}\sum_{k\in K} c_{ok}\cdot\tilde{y}_{ok\varepsilon}\cdot\gamma_{ok} - \sum_{d\in D_k}\sum_{k\in K} c_{dk}\cdot\tilde{u}_{dk\varepsilon}\cdot\tau_{dk} = z(\mathcal{DS}(\tilde{z}_\varepsilon,\tilde{y}_\varepsilon,\tilde{u}_\varepsilon)),
\end{aligned}
\tag{1}
$$

where $(\tilde{z}^0,\tilde{y}^0,\tilde{u}^0)\in ri(\mathbb{Q})$. Constraint (1) enforces that the optimal solution of $\mathcal{PO}_\varepsilon$ is chosen from the set of optimal solutions of dual subproblems $\mathcal{DS}_\varepsilon$. We note that $\mathcal{PO}_\varepsilon$ can be separated to $|K|$ independent subproblems $(\mathcal{PO}_{k\varepsilon})$, one for each $k\in K$. We thus have

$$
\begin{aligned}
(\mathcal{PO}_{k\varepsilon}):\quad & \max\left\{\alpha_k - \sum_{h\in H}\tilde{z}^0_{h\varepsilon}\cdot\theta_{hk} - \sum_{o\in O_k} c_{ok}\cdot\tilde{y}^0_{ok\varepsilon}\cdot\gamma_{ok} - \sum_{d\in D_k} c_{dk}\cdot\tilde{u}^0_{dk\varepsilon}\cdot\tau_{dk}\right\}\\
& \text{subject to:}\\
& \alpha_k - \sum_{h\in H}\tilde{z}_{h\varepsilon}\cdot\theta_{hk} - \sum_{o\in O_k} c_{ok}\cdot\tilde{y}_{ok\varepsilon}\cdot\gamma_{ok} - \sum_{d\in D_k} c_{dk}\cdot\tilde{u}_{dk\varepsilon}\cdot\tau_{dk} = z(\mathcal{DS}_k(\tilde{z}_\varepsilon,\tilde{y}_\varepsilon,\tilde{u}_\varepsilon)),\\
& \alpha_k - \theta_{e_1k} - \theta_{e_2k} - w_k\cdot\gamma_{ok} - w_k\cdot\tau_{dk}\leq t_{odek},\quad \forall\ o\in O_k,\ d\in D_k,\ e\in E_{odk},\ |e|=2,\\
& \alpha_k - \theta_{e_1k} - w_k\cdot\gamma_{ok} - w_k\cdot\tau_{dk}\leq t_{odek},\quad \forall\ o\in O_k,\ d\in D_k,\ e\in E_{odk},\ |e|=1,
\end{aligned}
$$

$$\theta_{hk} \geq 0, \quad \forall\, h \in H,$$
$$\gamma_{ok} \geq 0,\ \tau_{dk} \geq 0, \quad \forall\, o \in O_k,\ d \in D_k,$$

To generate a Pateto-optimal cut, any core point $(\tilde{z}^0, \tilde{y}^0, \tilde{u}^0) \in ri(\mathbb{Q})$ can be taken. We use the technique proposed in Papadakos (2008) to identify a core point that lies within the relative interior of the master problem polyhedron.

2. Training and Outputs for Logistic Regression Models

Recall that in Section 4.4 of the paper, learning-empowered elimination tests and variable reduction are introduced to reduce the size of the master problem and subproblems. In this section, we provide details on the training sample generation and the machine learning model outputs that were used in the numerical experiments reported in Section 5 of the paper.

More specifically, we generated three sets (S1, S2, and S3) of small-sized instances for training machine learning (i.e., logistic regression) models. The sizes of $(|O|, |D|, |K|, |H|)$ for sets S1, S2, and S3 were set to (15, 10, 25, 15), (20, 20, 20, 20), and (25, 25, 30, 25), respectively. There are 100 instances generated for each setting, leading to 300 instances for each set. The solution values of these small-sized instances produce 6,000, 22,636, and 22,576 records of $(\hat{z}, z_{cr}, z^{\varepsilon}_{rcr}, z_{LP}, z_{Lag}, z^{\varepsilon}_{BD})$, $(\hat{y}, y_{cr}, y^{\varepsilon}_{rcr}, y_{LP}, y_{Lag}, y^{\varepsilon}_{BD})$, and $(\hat{u}, u_{cr}, u^{\varepsilon}_{rcr}, u_{LP}, u_{Lag}, u^{\varepsilon}_{BD})$ for itertion ε, respectively, which were used to build the logistic regression models for computing $\mathcal{L}(z)$, $\mathcal{L}(y)$, and $\mathcal{L}(u)$. The models were fitted using the statsmodels.logit module in Python, with 70% data used for training and 30% data used for testing.

Figures 1–3 illustrate the relationships between $(\hat{z}_{avg}, \hat{y}_{avg}, \hat{u}_{avg})$ and $((z_{rc}, z_{rcr}, z_{BD}, z^{bin}_{LP}, z^{bin}_{Lag})$, $(y_{rc}, y_{rcr}, y_{BD}, y^{bin}_{LP}, y^{bin}_{Lag})$, $(u_{rc}, u_{rcr}, u_{BD}, u^{bin}_{LP}, u^{bin}_{Lag}))$ for the 5^{th} iteration, respectively. The values $(\hat{z}_{avg}, \hat{y}_{avg}, \hat{u}_{avg})$ are the average of the optimal solution values $(\hat{z}, \hat{y}, \hat{u})$ corresponding to the independent variables. From the figures, we observe that there is a strong negative correlation between $(\hat{z}_{avg}, \hat{y}_{avg}, \hat{u}_{avg})$ and $((z_{rc}, z_{rcr})$, (y_{rc}, y_{rcr}), $(u_{rc}, u_{rcr}))$ and a positive correlation between $(\hat{z}_{avg}, \hat{y}_{avg}, \hat{u}_{avg})$ and $((z_{BD}, z^{bin}_{LP}, z^{bin}_{Lag})$, $(y_{BD}, y^{bin}_{LP}, y^{bin}_{Lag})$, $(u_{BD}, u^{bin}_{LP}, u^{bin}_{Lag}))$. Further, Figure 4 displays the values of $\hat{z}_{avg}$ over the first 20 iterations of the Benders algorithm for each ranking of z_{rcr}. The plots show that the correlation between $\hat{z}_{avg}$ and z_{rcr} is significant and consistent over iterations of the Benders algorithm. These patterns are also observed for the correlation between $\hat{z}_{avg}$ and solution values of subproblems for the Benders algorithm and Lagrangian relaxation. Because the sizes of ranks (z_{rc}, z_{rcr}) are different for problems with different hub numbers, we transform (z_{rc}, z_{rcr}) to percentiles denoted by $(z^{p}_{cr}, z^{p}_{rcr})$, with $(z^{p}_{cr}, z^{p}_{rcr})$ = (round($\frac{z_{cr}}{|H|}$), round($\frac{z_{rcr}}{|H|}$)). Note that since the relationship between $\hat{z}_{avg}$ and $(z^{p}_{cr}, z^{p}_{rcr})$ is nonlinear, we set $(z^{p}_{cr}, z^{p}_{rcr})$ to 0.5 when they are bigger than 0.5.

We use variables $(z^{p}_{rc}, z^{p}_{rcr}, z_{BD}, z_{LP}, z_{Lag})$, $(y_{rc}, y_{rcr}, y_{BD}, y_{LP}, y_{Lag})$, and $(u_{rc}, u_{rcr}, u_{BD}, u_{LP}, u_{Lag})$ to build the logistic regression models for obtaining the classifiers for $\hat{z} = 1$, $\hat{y} = 1$, and $\hat{u} = 1$, respectively. The model outputs are presented in Tables 1–3, along with their ROC (receiver operating characteristic) curves and confusion matrices displayed in Figure 5 and Table 4, respectively. From these results, we observe that the problem size is not a statistically significant independent variable, implying that we can potentially transfer the knowledge learned from small instances to larger ones. When the size of the FHLP becomes bigger, obtaining the LP-relaxed solution values may become more challenging. For such cases, we should refit the logistic regression models by removing (z_{LP}, y_{LP}, u_{LP}) and update the model coefficients for the other independent variables.

References

Magnanti, T. L., R. T. Wong. 1981. Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research* **29** 464–484.

Papadakos, N. 2008. Practical enhancements to the magnanti-wong method. *Operations Research Letters* **36** 444–449.

Table 1 **Model Outputs for $\hat{z}=1$ Classifier**

C_c	Variable	coef	std err	z	$P>\|z\|$	[0.025	0.975]
0.7	Intercept	-7.0512	0.758	-9.299	0	-8.537	-5.565
	C(z_{BD}, Treatment(reference=0))[T.1]	0.6022	0.21	2.866	0.004	0.19	1.014
	z_{LP}	11.4837	0.735	15.624	0	10.043	12.924
	z_{Lag}	4.2078	0.935	4.499	0	2.375	6.041
	z^p_{cr}	-6.0689	0.716	-8.475	0	-7.472	-4.665
1.0	Intercept	-7.6978	0.965	-7.978	0	-9.589	-5.807
	z_{LP}	6.611	0.35	18.863	0	5.924	7.298
	z_{Lag}	6.2654	1.122	5.586	0	4.067	8.464
	z^p_{rcr}	-6.0057	0.823	-7.296	0	-7.619	-4.392
1.1	Intercept	-5.2515	0.553	-9.492	0	-6.336	-4.167
	C(z_{BD}, Treatment(reference=0))[T.1]	1.3408	0.447	2.999	0.003	0.465	2.217
	z_{LP}	11.7333	0.945	12.421	0	9.882	13.585
	z^p_{cr}	-4.2983	1.362	-3.157	0.002	-6.967	-1.629

Table 2 **Model Outputs for $\hat{y}=1$ Classifier**

C_c	Variable	coef	std err	z	$P>\|z\|$	[0.025	0.975]
0.7	Intercept	-2.1082	0.152	-13.875	0	-2.406	-1.81
	C(y_{BD}, Treatment(reference=0))[T.1]	0.9726	0.07	13.845	0	0.835	1.11
	y_{LP}	4.21	0.137	30.747	0	3.942	4.478
	y_{Lag}	2.9547	0.091	32.536	0	2.777	3.133
	y_{rcr}	-0.1164	0.044	-2.618	0.009	-0.203	-0.029
	y_{cr}	-0.229	0.036	-6.449	0	-0.299	-0.159
1.0	Intercept	-3.8608	0.06	-63.833	0	-3.979	-3.742
	y_{LP}	3.2268	0.065	49.424	0	3.099	3.355
	y_{Lag}	4.7146	0.093	50.787	0	4.533	4.897
1.1	Intercept	-5.1002	0.195	-26.15	0	-5.482	-4.718
	C(y_{BD}, Treatment(reference=0))[T.1]	1.4316	0.119	12.034	0	1.198	1.665
	y_{LP}	7.2304	0.146	49.528	0	6.944	7.516
	y_{Lag}	3.5245	0.19	18.557	0	3.152	3.897
	y_{rcr}	-0.1682	0.067	-2.516	0.012	-0.299	-0.037

Table 3 **Model Output for $\hat{u}=1$ Classifier**

C_c	Variable	coef	std err	z	$P>\|z\|$	[0.025	0.975]
0.7	Intercept	-2.3378	0.163	-14.355	0	-2.657	-2.019
	C(u_{BD}, Treatment(reference=0))[T.1]	1.1195	0.073	15.402	0	0.977	1.262
	u_{LP}	4.3046	0.14	30.737	0	4.03	4.579
	u_{Lag}	2.9716	0.094	31.452	0	2.786	3.157
	u_{rcr}	-0.1575	0.047	-3.373	0.001	-0.249	-0.066
	u_{cr}	-0.1369	0.038	-3.646	0	-0.21	-0.063
1.0	Intercept	-3.9274	0.062	-63.311	0	-4.049	-3.806
	u_{LP}	3.2192	0.065	49.433	0	3.092	3.347
	u_{Lag}	4.7532	0.094	50.74	0	4.57	4.937
1.1	Intercept	-5.6457	0.123	-45.942	0	-5.887	-5.405
	C(u_{BD}, Treatment(reference=0))[T.1]	1.4398	0.112	12.887	0	1.221	1.659
	u_{LP}	7.3576	0.149	49.527	0	7.066	7.649
	u_{Lag}	3.8466	0.186	20.634	0	3.481	4.212

Table 4 **Confusion Matrix**

C_c	$\hat{z}=1$ classifier			$\hat{y}=1$ classifier			$\hat{u}=1$ classifier		
	n=1800	Predicted: 0	Predicted: 1	n=6791	Predicted: 0	Predicted: 1	n=6773	Predicted: 0	Predicted: 1
0.7	Actual: 0 1,642	1,622	20	Actual: 0 2,292	1,944	348	Actual: 0 2,257	1,884	373
	Actual: 1 158	43	115	Actual: 1 4,499	313	4,186	Actual: 1 4,516	339	4,177
1.0	Actual: 0 1,661	1,647	14	Actual: 0 4,348	3,991	357	Actual: 0 4,288	3,950	338
	Actual: 1 139	19	120	Actual: 1 2,443	324	2,119	Actual: 1 2,485	332	2,153
1.1	Actual: 0 1,662	1,654	8	Actual: 0 4,572	4,461	111	Actual: 0 4,494	4,394	100
	Actual: 1 138	6	132	Actual: 1 2,219	111	2,108	Actual: 1 2,279	111	2,168

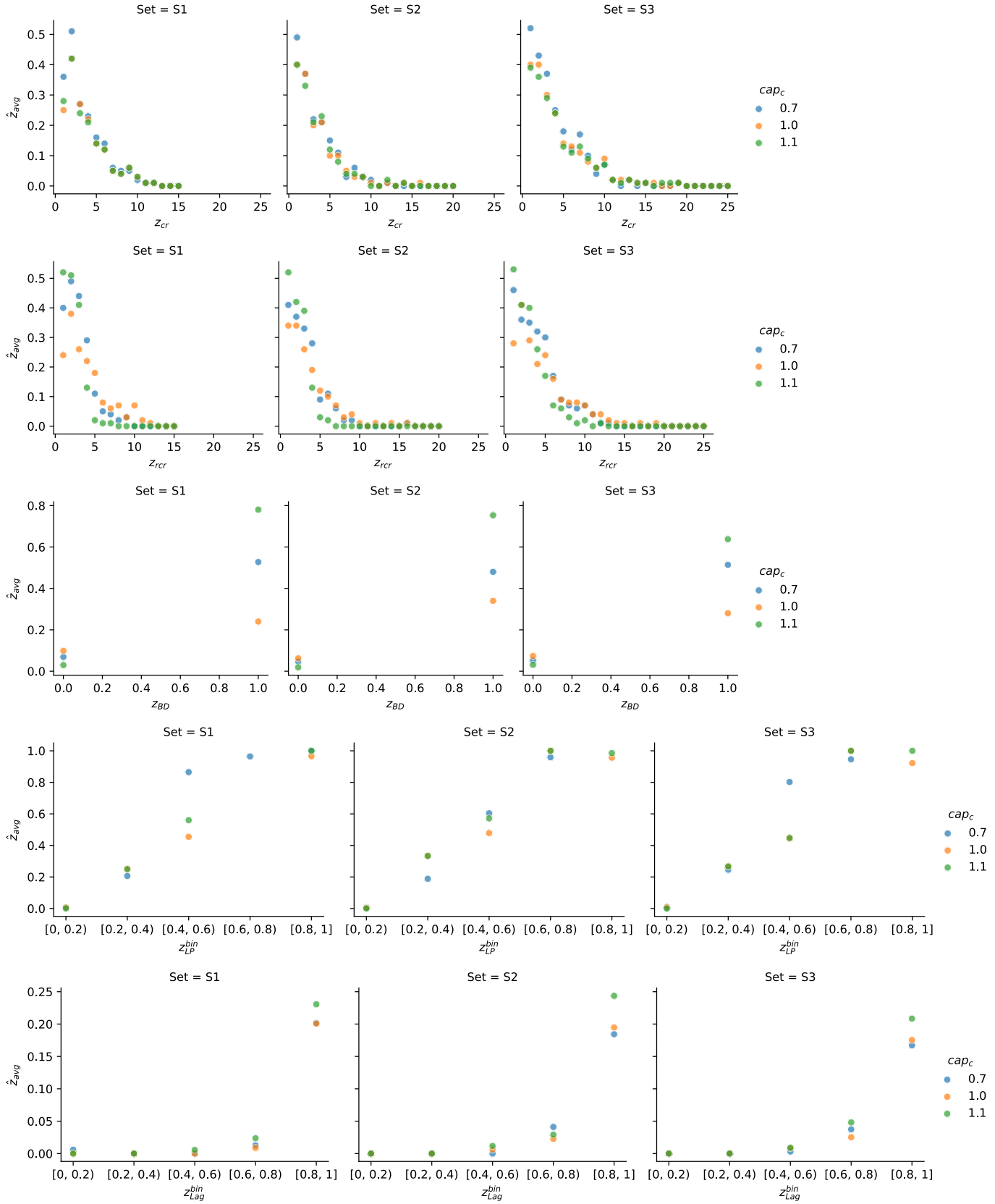

Figure 1 **Relationships between $\hat{z}_{avg}$ and $(z_{rc}, z_{rcr}, z_{BD}, z_{LP}^{bin}, z_{Lag}^{bin})$**

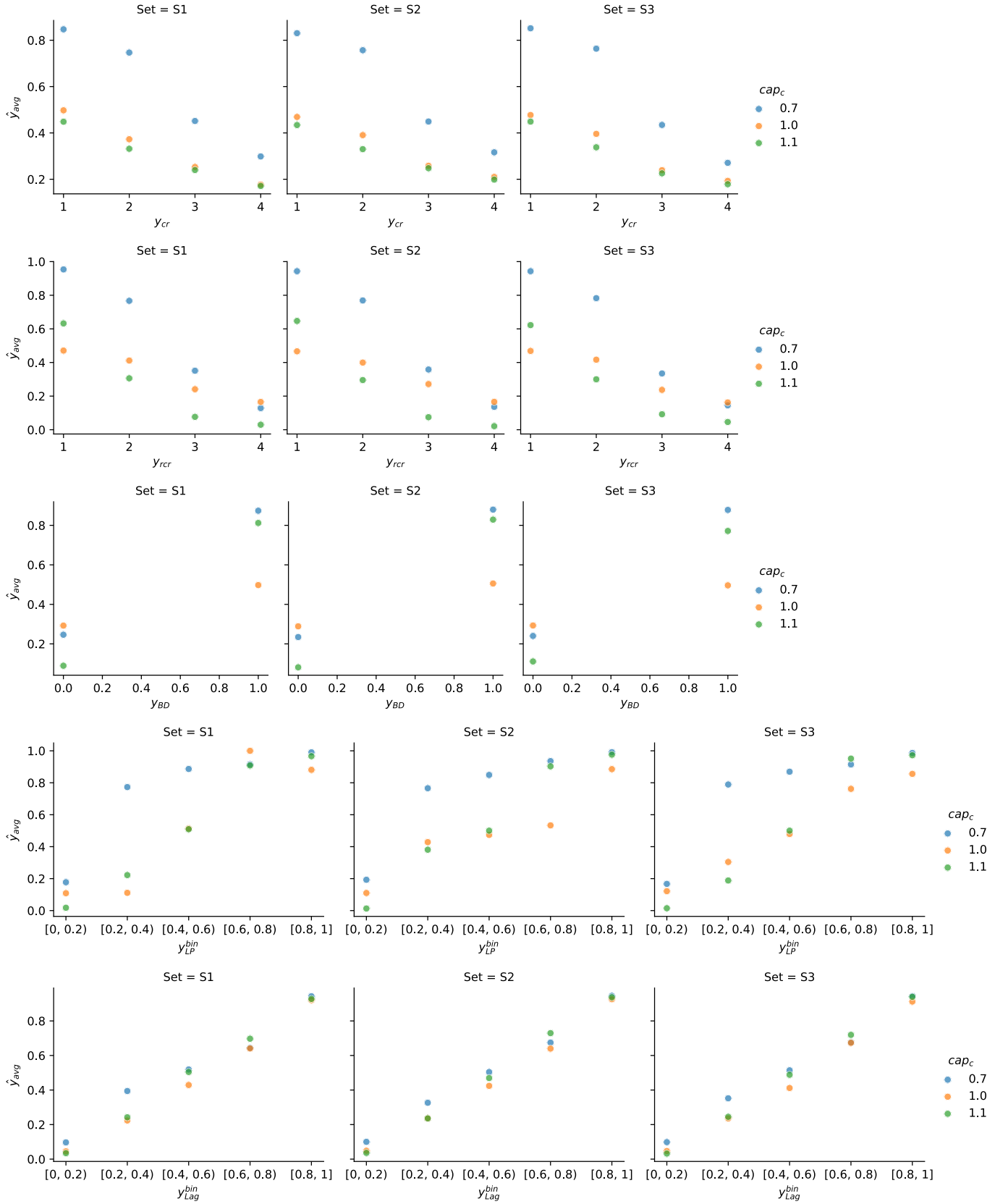

Figure 2 **Relationships between $\hat{y}_{avg}$ and $(y_{rc}, y_{rcr}, y_{BD}, y_{LP}^{bin}, y_{Lag}^{bin})$**

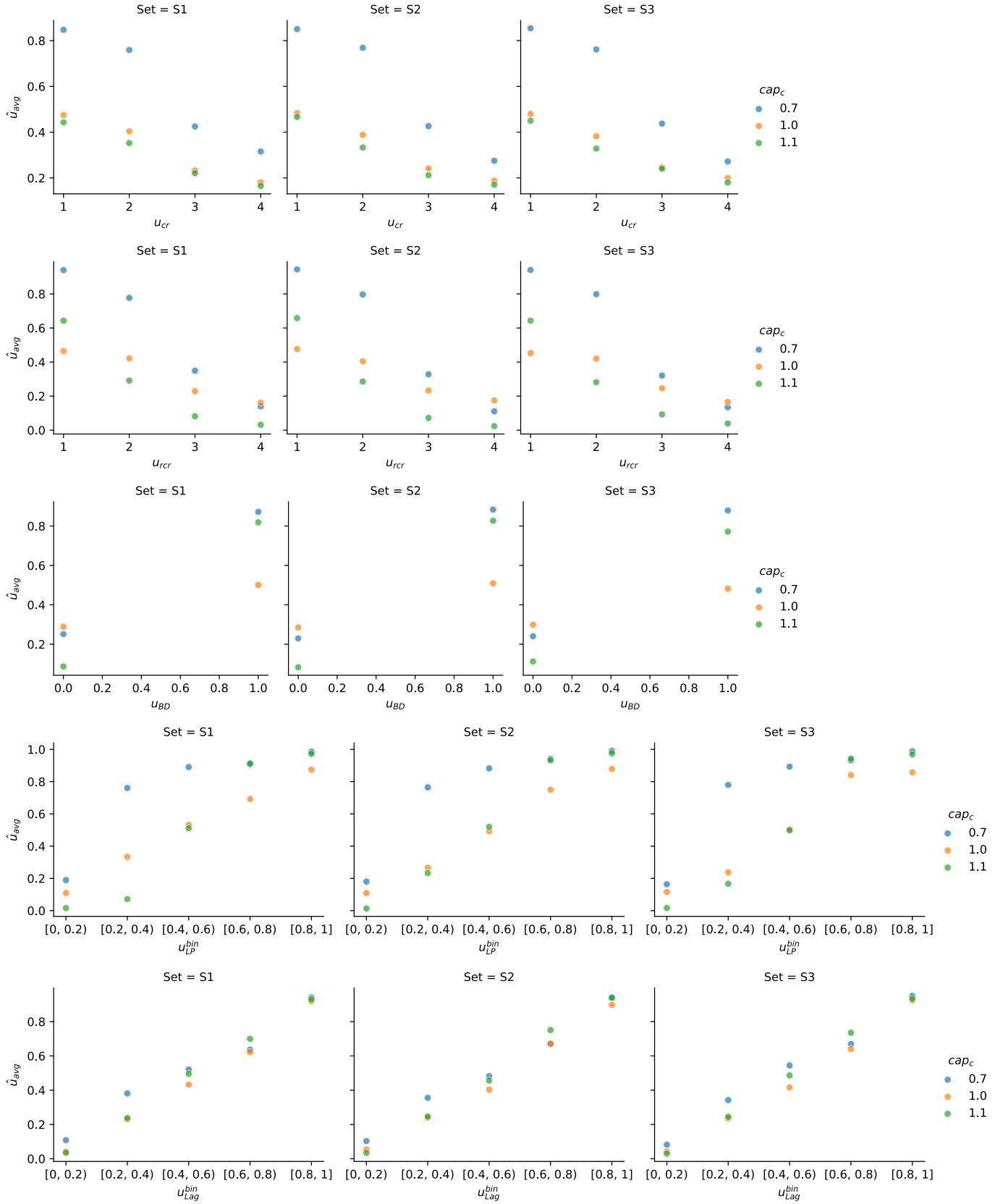

Figure 3 **Relationships between $\hat{u}_{avg}$ and $(u_{rc}, u_{rcr}, u_{BD}, u_{LP}^{bin}, u_{Lag}^{bin})$**

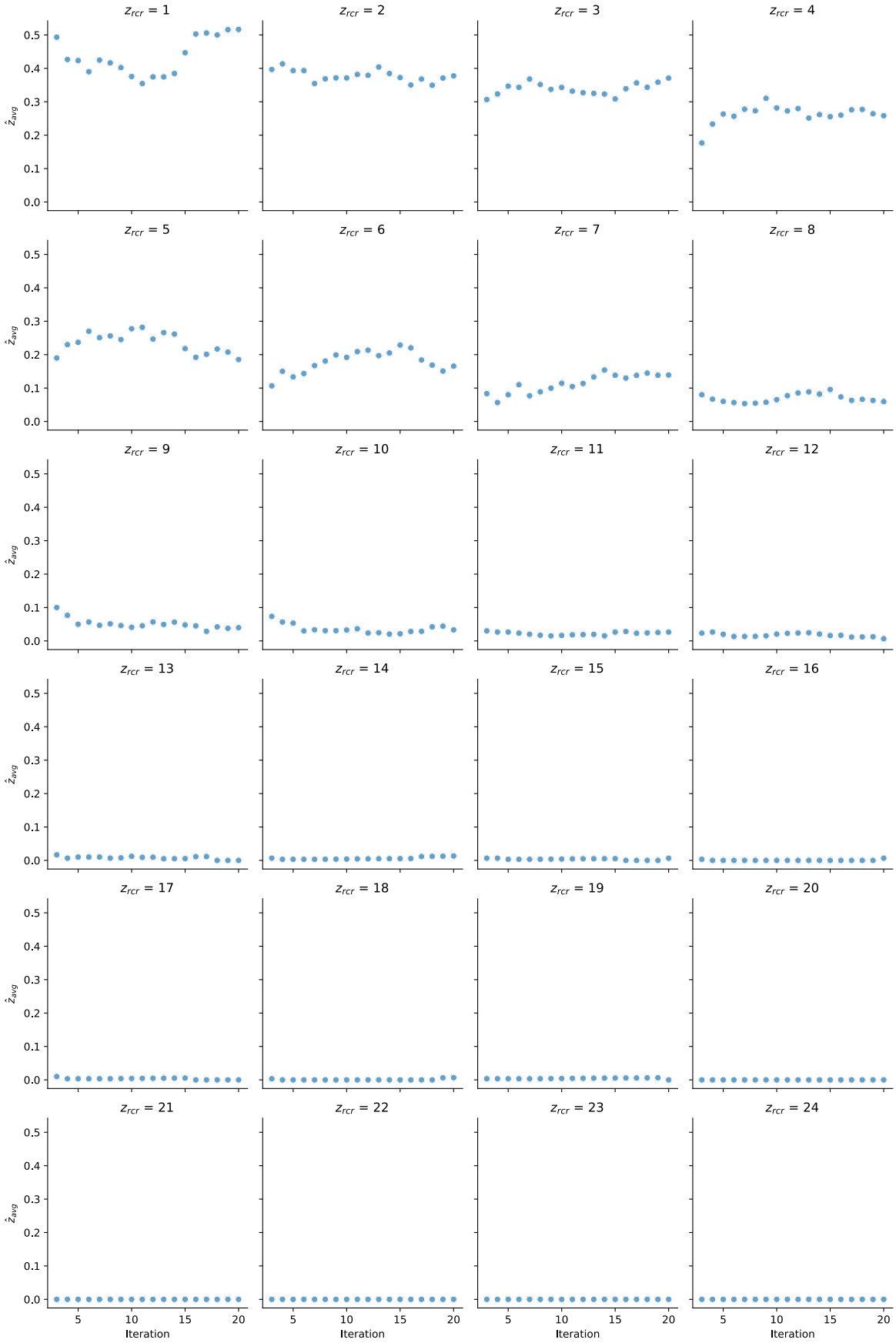

Figure 4 **Relationships between $\hat{u}_{avg}$ and u_{rcr} over Iterations of the Benders Algorithm**

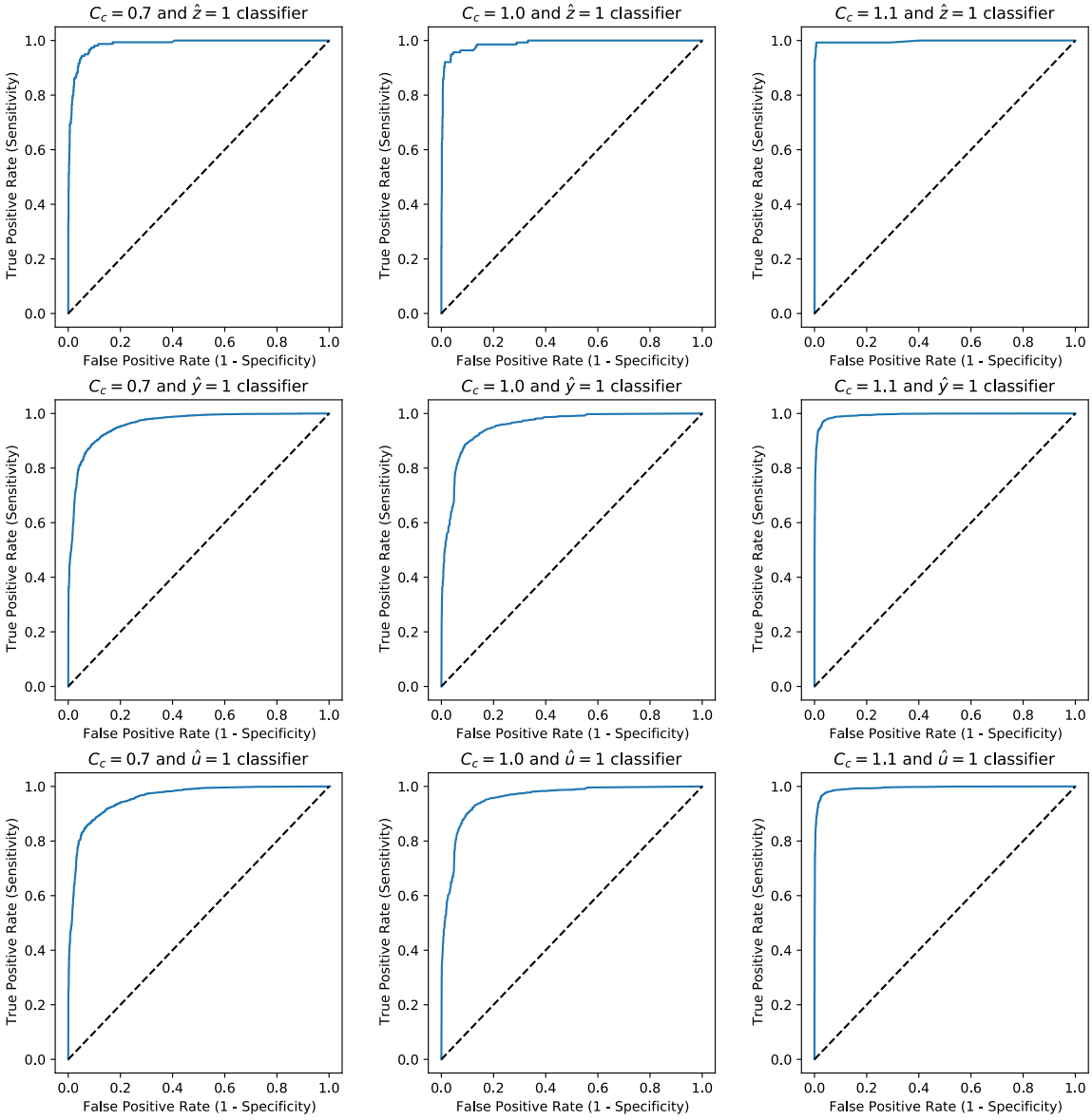

Figure 5 **ROC Curve**